

“UNIVERSIDAD NACIONAL DE PIURA”
FACULTAD DE CIENCIAS
ESCUELA PROFESIONAL DE INGENIERÍA ELECTRÓNICA
Y TELECOMUNICACIONES



**“SISTEMA DE MONITOREO DE OPERACIÓN CON
TECNOLOGIA GSM/GPS Y AHORRO DE ENERGIA PARA
CONTENEDORES DE PRODUCTO CONGELADOS (REFEER)”.**

**PRESENTADA POR:
BACH. JEISON IVAN ALVARADO CORDOVA**

**TESIS PARA OBTAR EL TÍTULO PROFESIONAL DE INGENIERO
ELECTRÓNICO Y TELECOMUNICACIONES**

**PIURA – PERÚ
2015**

**TESIS PRESENTADA COMO REQUISITO PARA OPTAR EL TÍTULO DE
INGENIERO ELECTRÓNICO Y TELECOMUNICACIONES**

ASESOR:



ING. FRANKLIN BARRA ZAPATA

TESISTA:



BACH. JEISON IVAN ALVARADO CORDOVA

JURADO EVALUADOR:

PRESIDENTE:



ING. AYAX MANUEL SIFUENTES MONTES

SECRETARIO:



ING. MIGUEL ANGEL PANDURO ALVARADO

VOCAL:



ING. ESTRADA CRISANTO-CESAR HUMBER

DEDICATORIA

A mis padres, hermano y abuelita por su apoyo incondicional durante mi formación profesional

AGRADECIMIENTO

Esta tesis no hubiera sido posible sin el impulso de todas aquellas personas que me han ayudado personalmente en el desarrollo del trabajo de esta índole, comenzando por el Ing. Franklin Barra Zapata a quien agradezco su apoyo en la elaboración del presente sistema de monitoreo.

Un sincero agradecimiento al jefe del área de trabajo, por darme permiso siempre que lo requería.

Un agradecimiento muy especial por la paciencia y el ánimo recibidos de mi familia y amigos.

A todos ellos, muchas gracias.

INDICE	PÁG.
ÍNDICE DE TABLAS	X
ÍNDICE DE FIGURAS	XI
ANEXOS	XIII
RESUMEN	XIV
ABSTRACT	XV
INTRODUCCIÓN	16

CAPÍTULO I

1.0 PLANTEAMIENTO DEL PROBLEMA	17
1.1 DELIMITACION	17
1.2 ENUNCIADO	18
1.3 APORTES DEL PROYECTO	18
1.4 HIPÓTESIS	18
1.5 OBJETIVOS	19
1.5.1 OBJETIVOS GENERALES	19
1.5.2 OBJETIVOS ESPECIFICOS	19
1.6 METODOLOGIA	19
1.6.1 TIPO Y NIVEL DE INVESTIGACION	19
1.6.1.1 TIPO DE INVESTIGACION	19
1.6.1.2 NIVEL DE INVESTIGACION	19
1.6.2 METODOLOGIA DE INVESTIGACION	20
1.6.2.1 METODO DE INVESTIGACION	20
1.6.2.2 DISEÑO DE LA INVESTIGACION	20
1.7 MATERIALES	20

CAPÍTULO II

2.0 MARCO TEÓRICO	21
2.1 MICROCONTROLADORES	21
2.1.1 ¿QUÉ ES UN MICROCONTROLADOR?	21
2.1.2 ¿QUE NO HACE EL MICROCONTROLADOR?	21
2.1.3 PRINCIPALES FABRICANTES	22
2.1.4 APARICIÓN Y DESARROLLO DE LOS MICROCONTROLADORES	23

2.1.4.1 BREVE ESBOZO HISTÓRICO	23
2.1.5 ARQUITECTURA DE UN MICROCONTROLADOR	24
2.1.5.1 ARQUITECTURA VON NEUMANN	24
2.1.5.2 ARQUITECTURA HARVARD	24
2.1.6 MICROCONTROLADOR A ELEGIR	25
2.1.6.1 ¿QUÉ MICROCONTROLADOR ELEGIR?	25
2.1.7 EL MICROCONTROLADOR 16F877A	26
2.1.7.1 CARACTERÍSTICAS PRINCIPALES	27
2.1.7.2 CARACTERÍSTICAS PERIFÉRICAS	27
2.1.7.3 CONFIGURACIÓN DE PINES	27
2.1.7.4 DESCRIPCIÓN DE LOS PINES DEL MICROCONTROLADOR	30
2.1.7.5 ARQUITECTURA INTERNA DEL MICROCONTROLADOR	31
2.1.8 MEMORIA DE DATOS (RAM)	33
2.1.8.1 RESUMEN DE ALGUNOS DE LOS REGISTROS DE CONFIGURACIÓN	34
2.2 TECNOLOGÍA GSM	35
2.2.1 ARQUITECTURA	35
2.2.2 DESARROLLO DE UNA LLAMADA GSM	38
2.2.3 VENTAJAS Y DESVENTAJAS ANTE OTRAS TECNOLOGÍA	39
2.2.3.1 VENTAJAS	39
2.2.3.2 DESVENTAJAS	39
2.2.4 LIMITACIONES DE GSM PARA LA TRANSMISIÓN DE DATOS	39
2.2.5 MODULO GPRS/GSM (CELULAR) SIM900	40
2.2.5.1 ESPECIFICACIONES TECNICAS	40
2.2.5.2 DIAGRAMA FISICO DEL HARDWARE Y DESCRIPCION	41
2.2.5.3 DIAGRAMA DE BLOQUES DEL MODULO SIM900	42
2.2.5.4 EVALUACION DEL SIM900	42
2.3. SISTEMA OPERATIVO ANDROID	43
2.3.1 ¿QUÉ ES ANDROID?	43
2.3.2 HISTORIA DE ANDROID	44
2.3.3 VERSIONES DISPONIBLES	44
2.3.4 ARQUITECTURA DE LA PLATAFORMA ANDROID	45

2.3.4.1 KERNEL	46
2.3.5 GUIA PARA DESARROLLADORES (BÁSICA)	46
2.3.6 PAUTAS PARA LAS INTERFACES DE USUARIO	48
2.3.7 APP INVENTOR	49
2.3.7.1 ¿PORQUÉ APP INVENTOR?	50
2.3.8 REQUERIMIENTOS DEL SISTEMA	50

CAPÍTULO III

3.0 DISEÑO DEL HARDWARE DEL SISTEMA MONITOREO Y AHORRO DE ENERGIA PARA REFEER	52
3.1 DIAGRAMA DE BLOQUES DEL SISTEMA DE MONITOREO GSM/GPS Y AHORRO DE ENERGÍA PARA CONTENEDORES DE PRODUCTOS CONGELADOS REFEER	52
3.2 DISEÑO DE LA TARJETA ELECTRÓNICA PARA SISTEMA DE MONITOREO GSM/GPS	53
3.2.1 MICROCONTROLADOR PIC16F877A	53
3.2.2 CIRCUITO PARA ON/OFF DEL CONTENEDOR.	54
3.2.3 VERIFICACIÓN DE ESTADO DEL CONTENEDOR ON-OFF	55
3.2.4 VISUALIZACIÓN LOCAL DE ESTADO DEL CONTENEDOR	57
3.2.5 CIRCUITO DE CONEXIÓN DEL GPS	58
3.2.6 CIRCUITO DE CONEXIÓN CON EL SENSOR DE TEMPERATURA DS18B20	59
3.2.7 ACTIVACIÓN DE ALARMAS	60
3.2.8 FUENTE DE ALIMENTACIÓN	60
3.2.9 COMUNICACIÓN CON EL MODEM GSM SIM900	61
3.3 DESCRIPCIÓN DEL CÓDIGO DE PROGRAMA DEL MICROCONTROLADOR	63
3.4 DESCRIPCIÓN DEL PROGRAMA PARA CELULAR	73
3.4.1 DESCRIPCIÓN DEL CÓDIGO DE PROGRAMA DE APLICACIÓN PARA ADMINISTRADOR	76
3.4.2 BOTÓN VER_INFO	77
3.4.3 BOTÓN TELEFONO1	77
3.4.4 BOTÓN TELEFONO2	77

3.4.5 BOTÓN TELEFONO3	78
3.4.6 BOTÓN TELEFONOS	79
3.4.7 BOTÓN ARMAR1	79
3.4.8 BOTÓN ARMAR2	79
3.4.9 BOTÓN ARMAR3	80
3.4.10 BOTÓN DESARMAR	80
3.4.11 BOTÓN AUTOMATICO	81
3.4.12 BOTÓN MANUAL	81
3.4.13 BOTÓN CNTONOFF.CLICK	82
3.4.14 BOTÓN CNTONOFF.LONGCLICK	82
3.4.15 BOTÓN ENVIAR_REF	83
3.4.16 BOTÓN proghora	83
3.4.17 BOTÓN FINALIZAR	84

CAPÍTULO IV

4.1 COSTO DE MATERIALES	85
4.2 COSTO DE MATERIALES MÁS INGENIERIA	86

CAPÍTULO V

5.0 CONCLUSIONES Y RECOMENDACIONES	87
5.1 CONCLUSIONES	87
5.2 RECOMENDACIONES	87
BIBLIOGRAFIA	88
ANEXOS	89

ÍNDICE DE TABLAS	PÁG
TABLA 2.0 CAPACIDAD MÁXIMA DE CORRIENTE TOTAL DE LOS PUERTOS DEL MICROCONTROLADOR	29
TABLA 2.1 DESCRIPCIÓN DE PINES DEL MICROCONTROLADOR 16F877A	30
TABLA 2.2 CONTINUACIÓN DE LA DESCRIPCIÓN DE PINES DEL MICROCONTROLADOR 16F877A	31
TABLA 2.3 CANALES DE CONTROL COMÚN UTILIZADOS POR LA ESTACIÓN BASE	36
TABLA 2.4 CANALES DE CONTROL DEDICADOS UTILIZADOS POR LA ESTACIÓN BASE	36
TABLA 3.1 TABLA DE COMANDOS POR SMS	62
TABLA 4.1 COSTO DE MATERIALES	85
TABLA 4.2 COSTO DE MATERIALES MAS INGENIERIA	86

ÍNDICE DE FIGURAS	PÁG.
FIG. 2.0 ARQUITECTURA VON NEUMANN	24
FIG. 2.1 ARQUITECTURA HARVARD	25
FIG. 2.2 DISTRIBUCIÓN DE PINES DEL PIC16F877A	28
FIG. 2.3 FOTOGRAFIA DEL PIC 16F877A	29
FIG. 2.4 DIAGRAMA DE BLOQUES DEL MICROCONTROLADOR 16F877A	32
FIG. 2.5 ORGANIZACIÓN DE LA MEMORIA RAM DEL PIC16F877	33
FIG. 2.6 ARQUITECTURA DEL SISTEMA GSM	37
FIG. 2.7 EJEMPLO DE UN ENRUTAMIENTO DE LLAMADAS GSM	38
FIG. 2.8 MODULO GPRS/GSM SIM 900	40
FIG. 2.9 DIAGRAMA FISICO DEL MODULO SIM 900	41
FIG. 2.10 DIAGRAMA LOGICO DEL SIM 900	42
FIG. 2.11 FOTOGRAFIA DEL MODULO SIM 900	43
FIG. 2.12 SISTEMAS DE CAPAS DE ANDROID	45
FIG. 2.13 ENTORNO DE DESARROLLO DE APP INVENTOR	49
FIG. 2.14 INTERFAZ DE APP INVENTOR 2	51
FIG. 2.15 USO DE EMULADOR VIRTUAL PARA PRUEBA DE LA APLICACIÓN	51
FIG. 3.1 SISTEMA DE MONITOREO Y AHORRO DE ENERGIA GSM/GPS	52
FIG. 3.2 TARJETA QUE SE ENCARGARA DEL MONITOREO DE TEMPERATURA Y ENERGIA ELECTRICA GSM/GPS	53
FIG. 3.3 MICROCONTROLADOR PIC16F877A CONEXIONES	54
FIG. 3.4 CIRCUITO ON/OFF DEL CONTENEDOR	55
FIG. 3.5 INTERRUPTOR DE 4 CONTACTOS	55
FIG. 3.6 RELÉ DE 11 PINES	56
FIG. 3.7 CIRCUITO DE CONEXIÓN POR RELÉ DE 11 PINES	56
FIG. 3.8 CONEXIÓN DEL RELÉ PARA MONITOREO DE FALTA DE ENERGÍA	57
FIG 3.9 VISUALIZACIÓN DE ESTADO DE CONTENEDORES	57
FIG. 3.10 CONEXIÓN DEL GPS CON MICROCONTROLADOR	58
FIG. 3.11 TRAMA DE DATOS TRANSMITIDO POR EL GPS	58
FIG. 3.12 CONEXIÓN DEL SENSOR DE TEMPERATURA DS18B20	59
FIG 3.13 SENSOR DE TEMPERATURA DS18B20	59
FIG. 3.14 CIRCUITO DE ACTIVACIÓN DE ALARMA/SIRENA	60

FIG. 3.15 FUENTE DE ALIMENTACIÓN DE LA TARJETA ELECTRÓNICA DEL SISTEMA	60
FIG. 3.16 MODEM GSM SIM900	61
FIG 3.17 CONEXIONES DEL MODEM GSM	61
FIG. 3.18 CONFIGURACIÓN DE LCD Y PUERTOS ANALÓGICOS COMO DIGITALES Y OTRAS VARIABLES	63
FIG. 3.19 VARIABLES DEL SENSOR DE TEMPERATURA Y HORA GMT EN EL GPS	64
FIG. 3.20 CONFIGURACIÓN DEL MODEM GSM SIM 900	64
FIG. 3.21 REGISTRO DE PROVEEDOR	65
FIG. 3.22 PROGRAMA PRINCIPAL	65
FIG. 3.23 COMANDO RING (RECEPCIÓN DE LLAMADA)	66
FIG. 3.24 PROGRAMACIÓN DE NÚMEROS DE CELULARES QUE RECIBIRÁN ALERTAS	66
FIG. 3.25 RUTINA PARA PROGRAMAR ALARMA POR TEMPERATURA	67
FIG. 3.26 RUTINA PARA ARMADO DE ALARMA	67
FIG. 3.27 RUTINA PARA DETERMINAR SI LA CLAVE ES CORRECTA	68
FIG. 3.28 RUTINA PARA ON / OFF LOS RELÉS QUE CONTROLAN EL INICIO O FIN DEL PROCESO EN CADA CONTENEDOR	68
FIG. 3.29 RUTINA DE HORA Y MINUTO PARA ENCENDIDO Y APAGADO DE CONTENEDOR	69
FIG. 3.30 RUTINA PARA LEER HORA DEL GPS Y ENCENDER O APAGAR CONTENEDOR	70
FIG. 3.31 RUTINA Q DETERMINA SI EL ENCENDIDO Y APAGADO DEL CONTENEDOR ES AUTOMÁTICO O MANUAL	70
FIGURA 3.32 RUTINA PARA ENVIÓ DE MENSAJES DE ALERTAS	71
FIG. 3.33 RUTINA ALARMA	71
FIGURA 3.34 RUTINA LEER HABILITACIÓN DE ALARMA Y VERIFICAR ON/OFF CONTENEDOR	72
FIG. 3.35. PARTE DEL CÓDIGO DE LECTURA DE TEMPERATURA DEL SENSOR DS18B20	72
FIG. 3.36. PANTALLAS DE LA APLICACIÓN	73
FIG. 3.37 VENTANA MODO DISEÑO DE LA APLICACIÓN	74
FIGURA 3.38 VENTANA MODO DISEÑO DE LA APLICACIÓN	74

FIG. 3.39 LISTA DE COMPONENTES DE LA APLICACIÓN	75
FIG. 3.40 NOMBRE DE COMPONENTES DE LA APLICACIÓN ADMINISTRADOR	76
FIG. 3.41 NOMBRE DE COMPONENTES DE LA APLICACIÓN ADMINISTRADOR	76
FIG. 3.42 BLOQUES DE LA APLICACIÓN PARA ADMINISTRADOR	76

RESUMEN

La presente tesis realiza el desarrollo de un sistema de monitoreo de contaminación ambiental producido por los vehículos de transporte público. Así mismo hace énfasis en la necesidad de contar con un sistema que nos permita monitorizar los gases que emiten los motores durante la combustión al ambiente y estén dentro del rango de valores asequibles para evitar daño a los seres humanos.

En el primer capítulo se describe la problemática y se establece el objetivo general y los objetivos específicos de la tesis, así como la justificación, los alcances y los límites de la misma. El segundo capítulo hace referencia al marco teórico y las definiciones que nos permiten entender desde la base la aplicación desarrollada.

En el tercer capítulo se muestra el desarrollo de la aplicación paso a paso tanto en el diseño del hardware como del software.

El cuarto capítulo nos permite analizar los resultados obtenidos al probar la aplicación

En el quinto capítulo se presentan las conclusiones a las que se llegaron con el desarrollo de la aplicación; y se enuncian las recomendaciones a seguir en el desarrollo de la aplicación.

PALABRAS CLAVE:

PBP (PIC BASIC PRO)

PIC (CONTROLADOR DE INTERFAZ PERIFÉRICO)

CO (MONÓXIDO DE CARBONO)

SIG (SPECIAL INTEREST GROUP)

ABSTRACT

The present thesis is accomplishes the development of a system of monitoring of environmental contamination produced by the public wagons. Likewise emphasis in the need makes of counting on a system that monitorizar allow us to the gases that emit the motors during the combustion to the environment and be within the range of accessible moral values to avoid damage to the human beings.

The problems is described in the first chapter and the general objective and the specific objectives of the thesis, as well as justification, are established catch up with them and the limits of the same. The second chapter makes reference to the theoretic frame and the definitions that they allow us to understanding from the base developed application.

The development of application is shown step by step so much in the hardware's design in the third chapter as if from the software.

The fourth chapter allows us examining the results obtained when trying the application

They encounter the findings they took place to with the development of the application in the fifth chapter; And they enunciate recommendations to follow in the development of application.

KEY WORDS

PBP (PIC BASIC PRO)

PIC (PERIPHERAL INTERFACE CONTROLLER)

CO (CARBON MONOXIDE)

SIG (GRUPO DE INTERESES ESPECIALES)

DISEÑO DE LA INVESTIGACIÓN

INTRODUCCIÓN

Un contenedor es un depósito de carga para el transporte aéreo, marítimo, fluvial, terrestre y multimodal. Son unidades que sirven de protección para las mercancías de la climatología y están fabricadas de acuerdo con la normativa ISO (por sus siglas en inglés "International Standardization Organization"), en concreto, ISO-6681; por ese motivo, también se conocen con el nombre de contenedores ISO.

Los contenedores son fabricados principalmente de acero, pero también hay de otros materiales como el de aluminio. Algunos son de madera contrachapada reforzados con fibra de vidrio. En la mayor parte de los casos, el suelo es de madera. Interiormente llevan un recubrimiento especial anti-humedad, para evitar las humedades durante el viaje. Otra característica que define a los contenedores es la presencia, en cada una de sus esquinas, de elementos que les permiten ser enganchados por grúas especiales, así como ser sujetos tanto en buques como en camiones.

Los más extendidos a nivel mundial son los equipos de 20 y 40 pies, con un volumen interno aproximado de 32,6 m³ y 66,7 m³ respectivamente. Las dimensiones de los contenedores están reguladas por la norma ISO 6346

Contenedores "Reefer" Cuentan con un sistema de producción y conservación de frío o calor y termostato. Deben ir conectados a un generador eléctrico externo; suelen funcionar bajo corriente trifásica

En los Contenedores "Reefer" la temperatura influye en el deterioro de los productos alimenticios perecederos y congelados más que ningún otro factor. Exponer productos frescos al calor o frío extremos puede provocar graves daños. Las temperaturas altas pueden desestabilizar la maduración y quemar el exterior de la fruta fresca y la verdura o bien reblandecerla. Las temperaturas bajas pueden provocar fácilmente la congelación y deshidratación de los productos o la aparición de orificios en el exterior de los mismos.

Además, las "temperaturas bajas" no siempre se sitúan alrededor de 0°C. Los plátanos son muy susceptibles de echarse a perder por el frío si la temperatura desciende por debajo de los 13°C.

Hay que tener en cuenta que estos contenedores son transportables vía terrestre, marítima o aérea.

Por lo que se pretende diseñar un sistema que este constantemente monitoreando el funcionamiento eléctrico verificando que este con corriente eléctrica.

ANTECEDENTES

Según H. Jiménez-Ariza, en su artículo titulado "REGISTRO Y ANÁLISIS DEL HISTORIAL TERMICO DURANTE EL ALMACENAMIENTO Y EL TRANSPORTE REFRIGERADO DE PRODUCTOS MÍNIMAMENTE PROCESADOS", indica lo siguiente: En este estudio se presenta el análisis de las temperaturas registradas durante el almacenamiento y el transporte refrigerado de un pequeño lote de hortalizas de hoja mínimamente procesadas, monitorizadas mediante tarjetas Turbotag y sensores inalámbricos Nlaza ubicados en el interior de las bolsas y en diferentes lugares de las cajas de cartón que contienen esas bolsas. El objetivo de este trabajo es generar protocolos de distribución y configuración de sensores y de procesamiento y análisis de los datos para el seguimiento y control continuo de la temperatura durante el transporte de productos refrigerados; se centra en el potencial de la utilización de la sensórica inalámbrica, evaluando la capacidad de los sensores en la obtención de datos y su comunicación, evaluando la cantidad de paquetes perdidos

Según CARLOS J RENEDO, en su artículo titulado "Trasporte Frigorífico", indica lo siguiente: La Cadena del Frío es el conjunto de elementos, medios y actividades necesarios para garantizar la calidad de un producto refrigerado hasta su consumo/utilización por el usuario final. Si se rompe la cadena del frío se reanuda la actividad bacteriana y se deterioran los productos.

El transporte es un eslabón esencial de la cadena de frío. Debe ser rápido, seguro y adaptable a las contingencias que puedan surgir. Los costes logísticos son determinantes para vender o no un producto. Se debe garantizar conservación propiedades productos de forma continua y eficaz. Una buena elección y diseño transporte frigorífico exige conocer características productos a manipular

1. PLANTEAMIENTO DEL PROBLEMA

1.1 DELIMITACION

Los contenedores refrigerados transportan artículos perecederos, y resulta clave mantener en ellos ciertos siempre bajo control algunos parámetros críticos, tales como la humedad y la temperatura dentro del contenedor.

Una supervisión inadecuada de este tipo de contenedores puede significar la pérdida de decenas o cientos de miles de dólares en un cargamento echado a perder. Sin embargo, supervisar manualmente los equipos de control que los contenedores suelen tener en su puerta, es una tarea engorrosa, costosa y sujeta a error.

El problema se ve agravado cuando los contenedores son movilizados y/o se van apilando.. Esta solución consiste en instalar en la puerta de cada contenedor refrigerado, donde tiene sus controles locales, una unidad remota formada por una tarjeta electrónica para verificar el funcionamiento eléctrico y un módulo de comunicación GSM/GPRS y geo-posicionamiento (GPS),. De esta forma, se puede controlar en todo momento y lugar la situación del contenedor, inclusive en pleno movimiento si es transportado vía terrestre por el Perú. También se propondrá una solución para transporte marítimo.

Esta tarjeta electrónica envía los estados de ON/OFF del Contenedor.

El resultado esperado es de minimizar los incidentes en los cargamentos, se automatiza la supervisión de los contenedores y se reducen así los costes operativos. También los importantes costes de las pólizas de seguros se ven reducidos, porque el servicio que presta la empresa transportadora es ahora más fiable y con menos riesgo. Y tal vez el valor añadido más importante: se fideliza a los usuarios en un mercado altamente competitivo.

1.2. ENUNCIADO

¿Es posible diseñar un sistema de monitoreo con tecnología GSM/GPS y ahorro de energía para contenedores de producto congelados?

1.3. APORTES DEL PROYECTO

- Incentivar la investigación y el uso de nuevos chips y sistemas de comunicación en aplicaciones de diseño de equipos y sistemas.
- Motivar el uso de nuevas tecnologías para el desarrollo de equipos y sistemas electrónicos.
- Demostrar que se puede solucionar problemas aplicando la electrónica e informática

1.4. HIPOTESIS

Usando las tecnologías existentes, si es posible diseñar un sistema de monitoreo con tecnología GSM/GPS y ahorro de energía para contenedores de producto congelados

1.5. OBJETIVOS

1.5.1. OBJETIVOS GENERAL

Diseñar y construir un sistema de monitoreo con tecnología GSM/GPS y ahorro de energía para contenedores de producto congelados

1.5.2. OBJETIVOS ESPECIFICOS

- Seleccionar elementos necesarios para detectar corte de energía eléctrica en el contenedor
- Diseñar hardware electrónico para monitoreo de los contenedores
- Diseñar hardware para visualización y avisos
- Diseñar y seleccionar hardware para sistema de Comunicación GSM y GPS
- Desarrollar Software del sistema

1.6. METODOLOGIA

1.6.1 Tipo y Nivel de la Investigación

1.6.1.1 Tipo de investigación:

De acuerdo con la investigación, la naturaleza del problema y el objetivo formulado, la investigación será tecnológica o de desarrollo. Según Velásquez y Rey este tipo de investigación tiene como finalidad la solución de problemas prácticos, lo cual implica intervención o modificación de la propia realidad; proceso éste que se manifiesta en el diseño de nuevos productos, procedimientos, métodos o metodologías, etc.

1.6.1.2 Nivel de investigación:

El nivel de investigación será descriptivo Según Hernández, Fernández y Baptista los estudios descriptivos pretenden medir o recoger información de manera independiente o conjunta sobre los conceptos o las variables a los que se refieren. Desde luego, pueden integrar las mediciones o información de cada una de dichas variables o conceptos para decir como es y cómo se manifiesta el fenómeno de interés; su objetivo no es indicar como se relacionan las variables medidas.

1.6.2 Metodología de la Investigación

1.6.2.1 Método de la Investigación

El desarrollo de la presente investigación se basará en el método de la ciencia o, simplemente, método científico; el mismo que está conformado por una serie de actividades correctamente articuladas que conducen a un resultado coherente con la propuesta de solución. Tales actividades son: problema nuevo para la ciencia, objetivos de la investigación, hipótesis de la investigación, variables, universo y muestra, técnicas e instrumentos, aplicación de los instrumentos, análisis de la información, contrastación de la hipótesis, formulación de conclusiones y recomendaciones.

1.6.2.2 Diseño de la Investigación

Por responder más apropiadamente a la naturaleza y tipo de Trabajo realizado, se ha seleccionado un diseño de Investigación pre experimental. Este diseño podría diagramarse de la siguiente manera:

G X O

Consiste en administrar un estímulo o tratamiento a un grupo y después aplicar una medición en una o más variables para observar cuál es el nivel del grupo en estas variables.

G Contenedores "Refeer" donde se la aplicara el estímulo X (prototipo) luego se medirá su eficiencia de las herramientas informática y electrónicas utilizadas y también la efectividad del nuevo proceso.

1.7. MATERIALES

- Relés de 24VAC
- Modulo GPS para envíos de SMS
- Microcontroladores
- Software: PIC BASIC, Microsoft Office 2007, Proteus 7.9 Profesional, App Inventor
- Internet, Correo Electrónico.
- Información técnica de circuitos integrados.

2.0 MARCO TEORICO

2.1 MICROCONTROLADORES

2.1.1 ¿QUÉ ES UN MICROCONTROLADOR?

El microcontrolador es un circuito integrado de muy alta escala de integración que contiene las partes funcionales de un computador:

- CPU (Central Processor Unit o Unidad de Procesamiento Central)
- Memorias volátiles (RAM), para datos
- Memorias no volátiles (ROM, PROM, EPROM) para escribir el programa
- Líneas de entrada y salida para comunicarse con el mundo exterior.
- Algunos periféricos (comunicación serial, temporizador, convertidor A/D, etc.)

“Es decir el microcontrolador es un computador integrado en un solo chip. Integrar todos estos elementos en un solo circuito integrado ha significado desarrollar aplicaciones importantes en la industria al economizar materiales, tiempo y espacio”. [1]

2.1.2 ¿QUE NO HACE EL MICROCONTROLADOR?

Las aplicaciones de un microcontrolador son tan inmensas que el límite es la propia imaginación del usuario. Estos microcontroladores están en el auto, en el televisor, en el teléfono, en una impresora, en un horno de microondas, en un juguete, en un transbordador espacial etc.

Los siguientes son algunos campos en los que los microcontroladores tienen gran uso:

- En la industria del automóvil: Control de motor, alarmas, regulador del servofreno, dosificador, etc.
- En la industria de los electrodomésticos: control de calefacciones, lavadoras, cocinas eléctricas, etc.
- En informática: como controlador de periféricos. Controlar impresoras, plotters, cámaras, scanners, terminales, unidades de disco, teclados, módems, etc.
- En la industria de imagen y sonido: tratamiento de la imagen y sonido, control de los motores de arrastre del giradiscos, magnetófono, video, etc.
- **En la industria, en general se utilizan en:**
 - Regulación: todas las familias de microcontroladores incorporan en alguna de sus versiones conversores A/D y D/A, para la regulación de la velocidad de las máquinas, de niveles, de temperatura, etc.
 - Automatismos: La enorme cantidad de líneas de entrada y salidas, y su inmunidad al ruido le hacen muy valioso para el control secuencial de procesos. Por ejemplo control de máquinas, herramientas, apertura y cierre automático de puertas según condiciones, plantas empaquetadoras, aparatos de maniobra de ascensores, etc.
 - Robótica: para control de los motores y captura de señales de los diferentes sensores, fabricación de controladores robóticos para sistemas automáticos, etc.

[1] <http://www.parallax.com/education>

- **Instrumentos portátiles compactos:**

- Radio paginador numérico (beeper)
- Planímetro electrónico
- Nivelímetro digital
- Identificador-probador de circuitos integrados
- Tacómetro digital
- Panel frontal de un osciloscopio
- Controlador de display LCD
- Analizador de espectros, etc.

- **Dispositivos autónomos:**

- Fotocopiadoras
- Máquinas de escribir
- Selector, Codificador decodificador de TV
- Localizador de peces
- Teléfonos de tarjeta
- Teléfonos celulares
- Cerraduras electrónicas
- Sistemas de seguridad

Se emplea también en medicina, en aplicaciones militares, edificios inteligentes, etc.

2.1.3 PRINCIPALES FABRICANTES

Por lo general los fabricantes de microprocesadores, lo son de microcontroladores. Los fabricantes de microcontroladores son más de 50, podemos mencionar a:

- ATMEL
- MOTOROLA
- INTEL
- MICROCHIP
- NEC
- HITACHI
- MITSUBISHI
- PHILIPS
- MATSUSHITA
- TOSHIBA
- AT&T
- ZILOG
- SIEMENS
- NATIONAL SEMICONDUCTOR, ETC. [2]

[2] <http://doru.info/projects/hdl/pavr/index.html>

2.1.4 APARICIÓN Y DESARROLLO DE LOS MICROCONTROLADORES

2.1.4.1 BREVE ESBOZO HISTÓRICO

La siguiente es una lista cronológica de los eventos tecnológicos más recientes que han tenido impacto sobre la aparición y el desarrollo del campo de los microcontroladores en la electrónica digital.

1971: Intel fabrica el primer microprocesador (el 4004) de tecnología PMOS. Este era un microprocesador de 4 bits y fue fabricado por Intel a petición de Datapoint Corporation con el objeto de sustituir la CPU de terminales inteligentes fabricadas en esa fecha por Datapoint mediante circuitería discreta. El dispositivo fabricado por Intel resultó 10 veces más lento de lo requerido y Datapoint no lo compró, de esta manera Intel comenzó a comercializarlo. El 4004 podía direccionar sólo 4096 (4k) posiciones de memoria de 4 bits, reconocía 45 instrucciones y podía ejecutar una instrucción en 20 µseg en promedio.

1972: Las aplicaciones del 4004 estaban muy limitadas por su reducida capacidad y rápidamente Intel desarrolló una versión más poderosa (el 8008), el cual podía manipular bytes completos, por lo cual fue un microprocesador de 8 bits. La memoria que este podía manejar se incrementó a 16 kbytes, sin embargo, la velocidad de operación continuó igual.

1973: Intel lanza al mercado el 8080 el primer microprocesador de tecnología NMOS, lo cual permite superar la velocidad de su predecesor (el 8008) por un factor de diez, es decir, el 8080 puede realizar 500K de operaciones por segundo, además se incrementó la capacidad de direccionamiento de memoria a 64 kbytes. A partir del 8080 de Intel se produjo una revolución en el diseño de microcomputadoras y varias compañías fabricantes de circuitos integrados comenzaron a producir microprocesadores. Algunos ejemplos de los primeros microprocesadores son: el IMP-4 y el SC/MP de National Semiconductors, el PPS-4 y PPS-8 de Rockwell International, el MC6800 de Motorola, el F-8 de Fairchild.

1975: Zilog lanza al mercado el Z80, uno de los microprocesadores de 8 bits más poderosos. En ese mismo año, Motorola reduce sus costos con sus microprocesadores 6501 y 6502 (este último adoptado por APPLE para su primera microcomputadora personal). Estos microprocesadores se comercializan en \$20 y \$25 (DLS USA) respectivamente. Esto provoca un auge en el mercado de microcomputadoras de uso doméstico y un caos en la proliferación de lenguajes, sistemas operativos y programas (ningún producto era compatible con el de otro fabricante).

1976: Surgen las primeras microcomputadoras de un solo chip, que más tarde se denominarán microcontroladores. Dos de los primeros microcontroladores, son el 8048 de Intel y el 6805R2 de Motorola.

198x: En la década de los 80's comienza la ruptura entre la evolución tecnológica de los microprocesadores y la de los microcontroladores, Ya que los primeros han ido incorporando cada vez más y mejores capacidades para las aplicaciones en donde se requiere el manejo de grandes volúmenes de información y por otro lado, los segundos han incorporado más capacidades que les permiten la interacción con el mundo físico en tiempo real, además de mejores desempeños en ambientes de tipo industrial.

2.1.5 ARQUITECTURA DE UN MICROCONTROLADOR

Según la arquitectura interna de la memoria de un microcontrolador se puede clasificar considerando como el CPU accede a los datos e instrucciones, en 2 tipos: [3]

2.1.5.1 ARQUITECTURA VON NEUMANN

Fue desarrollada por Jon Von Neumann, se caracteriza por tener una sola memoria principal donde se almacenan datos e instrucciones de forma indistinta. La CPU se conecta a través de un sistema de buses (direcciones, datos y control). Esta arquitectura es limitada cuando se demanda rapidez.

Memoria

Bus de direcciones Instrucciones

CPU +

Datos

Bus de datos

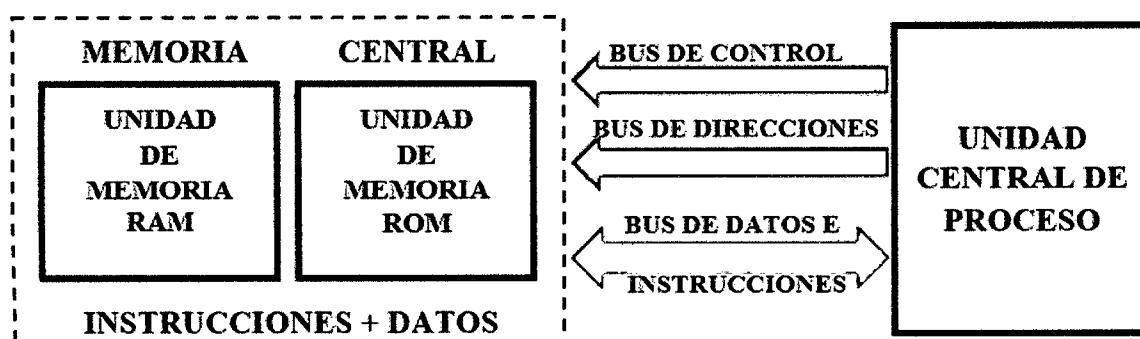


FIG. 2.0 ARQUITECTURA VON NEUMANN

2.1.5.2 ARQUITECTURA HARVARD

Fue desarrollado en Harvard, por Howard Aiken, esta arquitectura se caracteriza por tener 2 memorias independientes una que contiene sólo instrucciones y otra, que contiene sólo datos. Ambas, disponen de sus respectivos sistemas de buses para el acceso y es posible realizar operaciones de acceso simultáneamente en ambas memorias.

Existe una variante de esta arquitectura que permite el acceso a la tabla de datos desde la memoria de programas, es la Arquitectura de Harvard Modificada.

Esta última arquitectura es la dominante en los microcontroladores actuales ya que la memoria de programas es usualmente ROM, OTP, EPROM o FLASH, mientras que la memoria de datos es usualmente RAM.

Por ejemplo las tablas de datos pueden estar en la memoria de programa sin que sean perdidas cada vez que el sistema es apagado.

[3] Microcontroladores Pic, programación en basic. Carlos A. Reyes, 2da edic. pág. 18

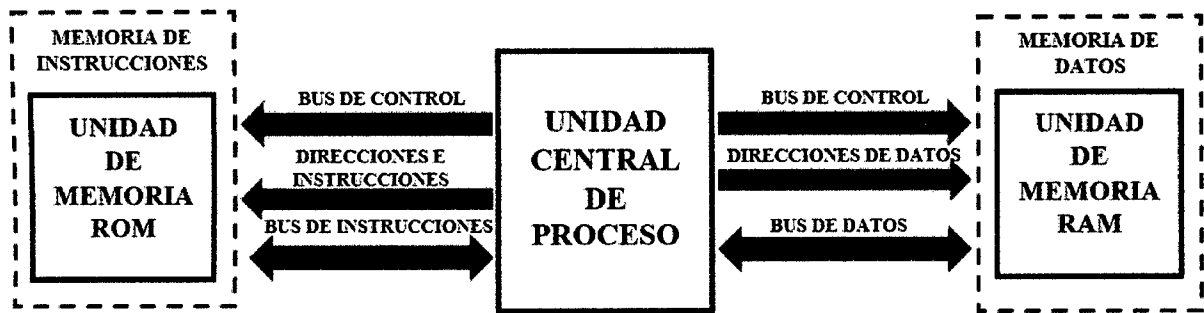


FIG. 2.1 ARQUITECTURA HARVARD

Se puede observar claramente que las principales ventajas de esta arquitectura son:

- El tamaño de las instrucciones no está relacionado con el de los datos, y por lo tanto puede ser optimizado para que cualquier instrucción ocupe una sola posición de memoria de programa, logrando así mayor velocidad y menor longitud de programa.
- El tiempo de acceso a las instrucciones puede superponerse con el de los datos, logrando una mayor velocidad de operación.

2.1.6 MICROCONTROLADOR A ELEGIR

2.1.6.1 ¿QUÉ MICROCONTROLADOR ELEGIR?

Sin duda la elección del microcontrolador dependerá de la tarea o proyecto que se tiene en mente pues los fabricantes como se mencionó anteriormente son más de 50, estos tienen muchos modelos enfocados a tareas específicas. Esta selección deberá ir de la mano con factores económicos óptimos así como de la idea del controlador incrustado (embedded controller), el cual es un controlador dedicado a una sola tarea e incorporado al sistema que gobierna.

- **PROCESAMIENTO DE DATOS:** Cuando se desea realizar cálculos complejos en un tiempo limitado, se debe seleccionar un microcontrolador suficientemente rápido para ello. Por otro lado, habrá que tener en cuenta la precisión de los datos a manejar: si no es suficiente con un microcontrolador de 8 bits, puede ser necesario acudir a microcontroladores de 16 ó 32 bits, o incluso a hardware de coma flotante.

- **ENTRADA/SALIDA:** Se debe identificar la cantidad y tipo de señales a controlar. Una vez realizado este análisis puede ser necesario añadir periféricos externos o cambiar a otro microcontrolador más adecuado a ese sistema.

- **CONSUMO:** algunos productos que incorporan microcontroladores están alimentados con baterías, puede ser que el microcontrolador esté trabajando en estado de bajo consumo pero debe “despertar” ante la activación de una señal (por ejemplo una interrupción) y ejecutar el programa adecuadamente.

- **MEMORIA:** para detectar las necesidades de memoria de una aplicación debemos saber la cantidad y el tipo de memoria necesaria, para esto se debe tener una versión preliminar (pseudo-código) de la aplicación y escoger el microcontrolador apropiado.

- **ANCHO DE PALABRA:** el criterio de diseño debe ser seleccionar el microcontrolador de menor ancho de palabra que satisfaga los requerimientos de la aplicación. Usar un microcontrolador de 4 bits supondrá reducir los costos, mientras que uno de 8 bits puede ser el más adecuado si el ancho de los datos es de un byte. Los microcontroladores de 16 y 32 bits, debido a su elevado costo, deben reservarse para aplicaciones que requieran altas prestaciones (Entrada/Salida grande o espacio de direccionamiento muy elevado).

- **DISEÑO DE LA PLACA:** la selección de un microcontrolador concreto condicionará el diseño de la placa de circuitos impresos.

2.1.7 EL MICROCONTROLADOR 16F877A

El microcontrolador PIC16F877 de Microchip pertenece a una gran familia de microcontroladores de 8 bits (bus de datos) que tienen las siguientes características generales que los distinguen de otras familias:

- Arquitectura Harvard
- Tecnología RISC
- Tecnología CMOS

Estas características se conjugan para lograr un dispositivo altamente eficiente en el uso de la memoria de datos y programa y por lo tanto en la velocidad de ejecución.

El PIC16F877 es un microcontrolador con memoria de programa tipo FLASH, lo que representa gran facilidad en el desarrollo de prototipos y en su aprendizaje ya que no se requiere borrarlo con luz ultravioleta como las versiones EPROM, sino que permite reprogramarlo nuevamente sin ser borrado con anterioridad.

“El PIC16F877 es un microcontrolador de tecnología Microchip fabricado en tecnología CMOS, su consumo de potencia es muy bajo y además es completamente estático, esto quiere decir que el reloj puede detenerse y los datos de la memoria no se pierden”. [4]

El encapsulado más común para este microcontrolador es el DIP (Dual In-line Pin) de 40 pines, propio para usarlo en experimentación. La referencia completa es PIC16F877-04 para el dispositivo que utiliza cristal oscilador de hasta 4 MHz, PIC16F877-20 para el dispositivo que utiliza cristal oscilador de hasta 20 MHz o PIC16F877A-I para el dispositivo tipo industrial que puede trabajar hasta a 20 MHz.

[4] <http://www.mikroe.com/chapters/view/84/libro>

Sin embargo, hay otros tipos de encapsulado que se pueden utilizar según el diseño y la aplicación que se quiere realizar. Por ejemplo, el encapsulado tipo surface mount (montaje superficial) tiene un reducido tamaño y bajo costo, que lo hace propio para producciones en serie o para utilizarlo en lugares de espacio muy reducido.

2.1.7.1 CARACTERÍSTICAS PRINCIPALES

- CPU RISC de alta performance
- Set de 35 instrucciones
Todas las instrucciones son de un ciclo salvo aquellas que incluyen saltos que son de 2 ciclos.
- Velocidad de Trabajo:
 - DC - 20 MHz de reloj de entrada
 - DC - 200 ns ciclo de instrucción
- Hasta 8K x 14 de trabajo y memoria FLASH de programación.
- Hasta 368 x 8 bytes, data de memoria (RAM)
- Hasta 256 x 8 bytes de Datos en Memoria EEPROM, manejo de Interrupciones (hasta 14 fuentes)
- Unidades aisladoras en hardware de 8 niveles
- Modo de direccionamiento directo, indirecto y relativo.
- Power-on Reset (POR)
- Power-up Timer (PWRT) y Oscilador Start-up Timer (OST)
- Watchdog Timer (WDT) con el reloj RC interno para mejor seguridad.
- Protección de código programable.
- Programación serial vía 2 pines, programación serial en el circuito (ICSP)
- In-Circuit Debugging vía 2 pines
- Amplio rango de voltaje de trabajo: 2.0V a 5.5V

2.1.7.2 CARACTERÍSTICAS PERIFÉRICAS

- Timer0: 8-bit timer/counter con 8-bit de pre-escala
- Timer1: 16-bit timer/counter con pre-escala, que puede ser incrementado durante el modo SLEEP vía reloj externo.
- Timer2: 8-bit timer/counter con registro de período de 8-bit, prescaler y postscaler
- Dos módulos de captura, comparativa, PWM
- Captura de 16-bit, Máx. Resolución: 12.5 ns
- Compara 16-bit, máx. Resolución: 200 ns
- PWM máx. Resolución: 10-bit
- Convertidor Analógico - Digital de 10-bit multicanal
- Puerto de sincronización serial (SSP) con SPI (Modo maestro) e I2C (Maestro/Esclavo)
- Universal Synchronous Asynchronous Receiver Transmitter (USART/SCI) con detección de direcciones de 9-bit
- Puerto paralelo esclavo (PSP) de 8-bits de ancho, con controles externos de RD, WR y CS (solo 40/44-pin)
- Brown-out detection circuitry para Brown-out Reset (BOR)

2.1.7.3 CONFIGURACIÓN DE PINES

Los pines de entrada/salida de este microcontrolador están organizados en cinco puertos, el puerto A con 6 líneas, el puerto B con 8 líneas, el puerto C con 8 líneas, el puerto D con 8 líneas y el puerto E con 3 líneas.

Cada pin de esos puertos se puede configurar como entrada o como salida independiente programando un par de registros diseñados para tal fin. En ese registro un bit en "0" configura el pin del puerto correspondiente como salida y un bit en "1" lo configura como entrada.

Dichos pines del microcontrolador también pueden cumplir otras funciones especiales, siempre y cuando se configuren para ello, según se verá más adelante.

Los pines del puerto A y del puerto E pueden trabajar como entradas para el convertidor Análogo a Digital interno, es decir, allí se podría conectar una señal proveniente de un sensor o de un circuito analógico para que el microcontrolador la convierta en su equivalente digital y pueda realizar algún proceso de control o de instrumentación digital. El pin RB0/INT se puede configurar por software para que funcione como interrupción externa, para configurarlo se utilizan unos bits de los registros que controlan las interrupciones.

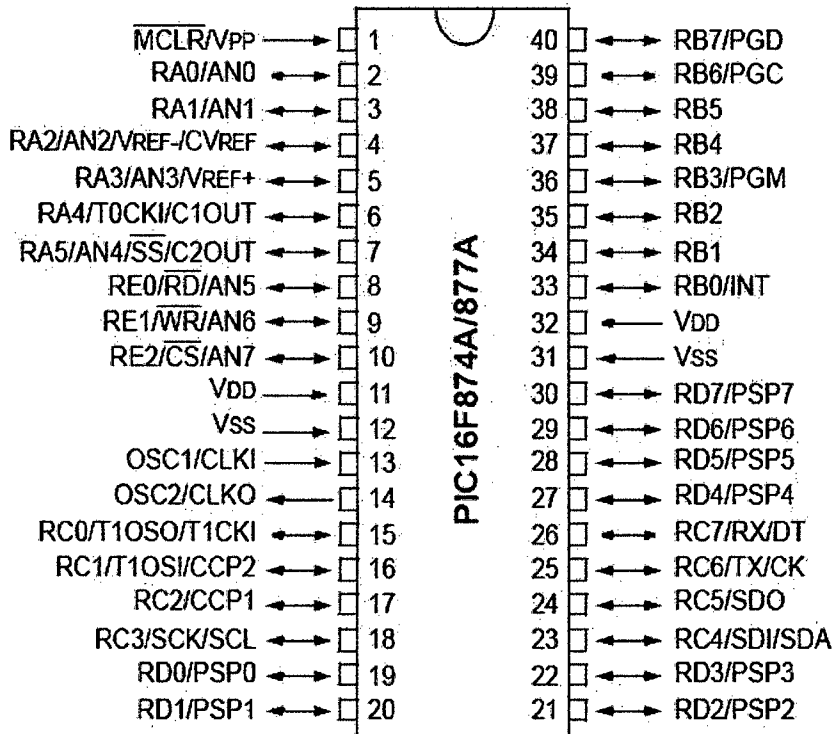


FIG. 2.2 DISTRIBUCIÓN DE PINES DEL PIC16F877A

El pin RA4/T0CKI del puerto A puede ser configurado como un pin de entrada/salida o como entrada del temporizador/contador. Cuando este pin se programa como entrada digital, funciona como un disparador de Schmitt (Schmitt trigger), puede reconocer señales un poco distorsionadas y llevarlas a niveles lógicos (cero y cinco voltios).

Cuando se usa como salida digital se comporta como colector abierto (open collector), por lo tanto, se debe poner una resistencia de pull-up (resistencia externa conectada a un nivel de cinco voltios).

Como salida, la lógica es inversa: un "0" escrito al pin del puerto entrega en el pin un "1" lógico. Además, como salida no puede manejar cargas como fuente, sólo en el modo sumidero.

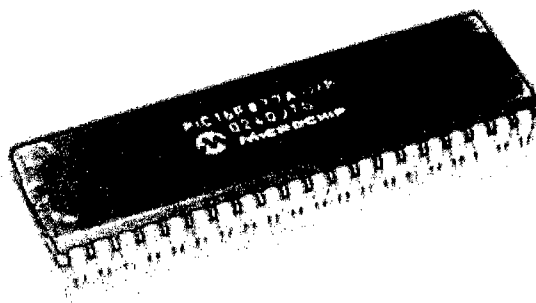


FIG. 2.3 FOTOGRAFIA DEL PIC 16F877A

	PUERTO A	PUERTO B	PUERTO C	PUERTO D
Modo sumidero	150 mA	200 mA	200 mA	200 mA
Modo fuente	150 mA	200 mA	200mA	200mA

TABLA 2.0 CAPACIDAD MÁXIMA DE CORRIENTE TOTAL DE LOS PUERTOS DEL MICROCONTROLADOR

El consumo de corriente del microcontrolador para su funcionamiento depende del voltaje de operación, la frecuencia y de las cargas que tengan sus pines. Para un oscilador de 4 MHz el consumo es de aproximadamente 2 mA; aunque este se puede reducir a 40 microamperios cuando se está en el modo sleep (en este modo el micro se detiene y disminuye el consumo de potencia).

2.1.7.4 DESCRIPCIÓN DE LOS PINES DEL MICROCONTROLADOR

NOMBRE PIN	PIN	DESCRIPCIÓN
RA0/AN0	2	E/S Digital o Entrada análoga 0.
RA1/AN1	3	E/S Digital o Entrada análoga 1.
RA2/AN2 Vref -	4	E/S Digital o Entrada análoga 2.
RA3/AN3/Vref +	5	E/S Digital o Entrada análoga 3.
RA4/T0CKI	6	Bit 4 del puerto A (E/S bidireccional). También se usa como entrada de reloj al temporizador/contador TMR0.
RA5/SS/AN4	7	E/S Digital o Entrada análoga 4. También lo usa el puerto serial síncrono.
RB0/INT	33	Bit 0 del puerto B (E/S bidireccional). Buffer E/S: TTL/ST. También se usa como entrada de interrupción externa (INT).
RB1	34	Bit 1 del puerto B (E/S bidireccional). Buffer E/S: TTL
RB2	35	Bit 2 del puerto B (E/S bidireccional). Buffer E/S: TTL
RB3/PGM	36	Bit 3 del puerto B (E/S bidireccional). Buffer E/S: TTL (Programación en bajo voltaje)
RB4	37	Bit 4 del puerto B (E/S bidireccional). Buffer E/S: TTL. Interrupción por cambio del pin.
RB5	38	Bit 5 del puerto B (E/S bidireccional). Buffer E/S: TTL. Interrupción por cambio del pin.
RB6/PGC	39	Bit 6 del puerto B (E/S bidireccional). Buffer E/S: TTL/ST. Interrupción por cambio del pin. Entrada de reloj para programación serial.
RB7/PGD	40	Bit 7 del puerto B (E/S bidireccional). Buffer E/S: TTL/ST. Entrada de datos para programación serial.
RC0/T1OSO/T1CKI	15	E/S Digital. Salida del oscilador Timer1 o entrada de reloj Timer1.
RC1/T1OSI/CCP2	16	E/S Digital. Entrada del oscilador Timer 1. Entrada Captura 2; Salida Compara 2; Salida PWM 2
RC2/CCP1	17	E/S Digital. Entrada Captura 1; Salida Compara 1; Salida PWM 1
RC3/SCK/SCL	18	E/S Digital. Línea de reloj serial asíncrono en el modo SPI y el modo I ² C
RC4/SDI/SDA	23	E/S Digital. Línea de datos en el modo SPI o en el modo I ² C
RC5/SDO	24	E/S Digital.
RC6/TX/CK	25	E/S Digital. Transmisión asíncrona (USART) o reloj síncrono (SSP).

TABLA 2.1 DESCRIPCIÓN DE PINES DEL MICROCONTROLADOR 16F877A

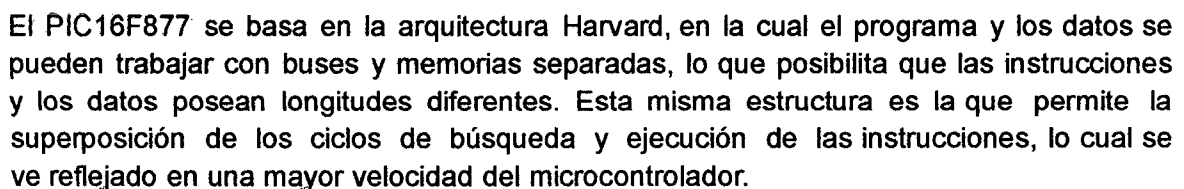
RC7/RX/DT	26	E/S Digital. Recepción asíncrona (USART) o línea de datos (SSP).
VDD	11,32	Voltaje de alimentación DC (+)
VSS	12,31	Referencia de voltaje (GND).
MCLR	1	Entrada de RESET al microcontrolador. Voltaje de entrada durante la programación. En nivel bajo resetea el microcontrolador.
OSC1/CLKIN	13	Entrada oscilador cristal oscilador / Entrada fuente de reloj externa.
OSC2/CLKOUT	14	Salida oscilador cristal. Oscilador RC: Salida con un ¼ frecuencia OSC1
RD0/PSP0	19	E/S Digital. Puede ser puerto paralelo en bus de 8 bits.
RD1/PSP1	20	E/S Digital. Puede ser puerto paralelo en bus de 8 bits.
RD2/PSP2	21	E/S Digital. Puede ser puerto paralelo en bus de 8 bits.
RD3/PSP3	22	E/S Digital. Puede ser puerto paralelo en bus de 8 bits.
RD4/PSP4	27	E/S Digital. Puede ser puerto paralelo en bus de 8 bits.
RD5/PSP5	28	E/S Digital. Puede ser puerto paralelo en bus de 8 bits.
RD6/PSP6	29	E/S Digital. Puede ser puerto paralelo en bus de 8 bits.
RD7/PSP7	30	E/S Digital. Puede ser puerto paralelo en bus de 8 bits.
RE0/RD/AN5	8	E/S Digital. Puede ser pin de lectura (read) en modo microprocesador.
RE1/WR/AN6	9	E/S Digital. Puede ser pin de escritura (write) en modo microprocesador.
RE2/CS/AN7	10	E/S Digital. Puede ser pin de selección de chip (chip select) en modo microprocesador.

TABLA 2.2 CONTINUACIÓN DE LA DESCRIPCIÓN DE PINES DEL MICROCON 16F877A
2.1.7.5 ARQUITECTURA INTERNA DEL MICROCONTROLADOR

Este término se refiere a los bloques funcionales internos que conforman el microcontrolador y la forma en que están conectados, por ejemplo la memoria FLASH (de programa), la memoria RAM (de datos), los puertos, la lógica de control que permite que todo el conjunto funcione, etc.

La figura 2.4 muestra la arquitectura general del PIC16F877, en ella se pueden apreciar los diferentes bloques que lo componen y la forma en que se conectan. Se muestra la conexión de los puertos, las memorias de datos y de programa, los bloques especiales como el Watchdog, los temporizadores de arranque, el oscilador, etc.

FIG. 2.4 DIAGRAMA DE BLOQUES DEL MICROCONTROLADOR 16F877A



2.1.8 MEMORIA DE DATOS (RAM)

El PIC16F877 posee cuatro bancos de memoria RAM, cada banco posee 128 bytes. De estos 128 los primeros 32 (hasta el 1Fh) son registros que cumplen un propósito especial en el control del microcontrolador y en su configuración. Los 96 siguientes son registros de uso general que se pueden usar para guardar los datos temporales de la tarea que se está ejecutando, figura 2.5 Todas las posiciones o registros de memoria se pueden acceder directa o indirectamente (esta última forma a través del registro selector FSR). Para seleccionar que página o banco de memoria se trabaja en un momento determinado se utilizan los bits RP0 y RP1 del registro STATUS.

INDF	00h	INDF	80h	INDF	100h	INDF	180h	
TMR0	01h	OPTION_REG	81h	TMR0	101h	OPTION_REG	181h	
PCL	02h	PCL	82h	PCL	102h	PCL	182h	
STATUS	03h	STATUS	83h	STATUS	103h	STATUS	183h	
FSR	04h	FSR TRISA	84h	FSR	104h	FSR	184h	
PORTA	05h	TRISA	85h		105h		185h	
PORTB	06h	TRISB	86h	PORTB	106h	TRISB	186h	
PORTC	07h	TRISC	87h		107h		187h	
PORTD	08h	TRISD	88h		108h		188h	
PORTE	09h	TRISE	89h		109h		189h	
PCLATH	0Ah	PCLATH	8Ah	PCLATH	10Ah	PCLATH	18Ah	
INTCON	0Bh	INTCON	8Bh	INTCON	10Bh	INTCON	18Bh	
PIR1	0Ch	PIE1	8Ch	EEDATA	10Ch	EECON1	18Ch	
PIR2	0Dh	PIE2	8Dh	EEADR	10Dh	EECON2	18Dh	
TMR1L	0Eh	PCON	8Eh	EEDATH	10Eh	Reservado	18Eh	
TMR1H	0Fh		8Fh	EEADRH	10Fh	Reservado	18Fh	
T1CON	10h		90h		110h		190h	
TMR2	11h	SSPCON2	91h	Registros de Propósito General 16 Bytes		Registros de Propósito General 16 Bytes		
T2CON	12h	PR2	92h					
SSPBUF	13h	SSPADD	93h					
SSPCON	14h	SSPSTAT	94h					
CCPR1L	15h		95h					
CCPR1H	16h		96h					
CCP1CON	17h		97h					
RCSTA	18h	TXSTA	98h					
TXREG	19h	SPBRG	99h					
RCREG	1Ah		9Ah					
CCPR2L	1Bh		9Bh					
CCPR2H	1Ch		9Ch					
CCP2CON	1Dh		9Dh					
ADRESH	1Eh	ADRESL	9Eh					
ADCON0	1Fh	ADCON1	9Fh					
	20h		A0h		11Fh		19Fh	
Registros de Propósito General 96 Bytes	7Fh	Registros de Propósito General 80 Bytes		Registros de Propósito General 60 Bytes		Registros de Propósito General 80 Bytes		
Banco 0		Banco 1		Banco 2		Banco 3		

FIG. 2.5 ORGANIZACIÓN DE LA MEMORIA RAM DEL PIC16F877

2.1.8.1 RESUMEN DE ALGUNOS DE LOS REGISTROS DE CONFIGURACIÓN

BANCO 0:

TMR0: Registro del temporizador/contador de 8 bits.

PCL: Byte menos significativo del contador de programa (PC).

STATUS: Contiene banderas (bits) que indican el estado del procesador después de una operación aritmética/lógica.

FSR: Registro de direccionamiento indirecto.

PORTA, PORTB, PORTC, PORTD, PORTE: Registro de puertos de E/S de datos. Conectan con los pines físicos del micro.

PCLATH: Byte alto (más significativo) del contador de programa (PC).

INTCON: Registro de control de las interrupciones.

ADRESH: Parte alta del resultado de la conversión A/D.

ADCON0: Controla la operación del módulo de conversión A/D

BANCO 1:

OPTION: Registro de control de frecuencia del TMR0.

TRISA, TRISB, TRISC, TRISD, TRISE: Registros de configuración de la operación de los pines de los puertos.

ADRESL: Parte baja del resultado de la conversión A/D.

ADCON1: Controla la configuración de los pines de entrada análoga.

BANCO 2:

TMR0: Registro del temporizador/contador de 8 bits.

PCL: Byte menos significativo del contador de programa (PC).

FSR: Registro de direccionamiento indirecto.

EEDATA: Registro de datos de la memoria EEPROM.

EEADR: Registro de dirección de la memoria EEPROM.

PCLATH: Byte alto (más significativo) del contador de programa (PC).

INTCON: Registro de control de las interrupciones.

BANCO 3:

OPTION: Registro de control de frecuencia del TMR0.

EECON1: Control de lectura/escritura de la memoria EEPROM de datos.

EECON2: No es un registro físico.

2.2 TECNOLOGÍA GSM

El sistema GSM (Global System for Mobile Communication) puede ser definido como un sistema de radio comunicaciones, porque está destinado para comunicaciones de media y larga distancia, digital, móvil y celular, lo interesante del sistema es que nos proporciona servicios de datos, e incluye integración de servicios.

“Este sistema nos proporciona una calidad de audio que ayuda a la fiabilidad de la comunicación, incluye sistemas de seguridad en el aspecto de autenticación de usuarios, actualmente este sistema es utilizado a nivel mundial por la mayoría de operadores celulares.”[5]

El roaming internacional es una de las armas poderosas con la que cuenta este sistema, especialmente utilizado por los usuarios de negocios que constantemente viajan al exterior, solamente con tener un móvil que trabaje en las cuatro bandas y un convenio internacional de operadoras el usuario accede de manera automática a este servicio.

El sistema GSM nos proporciona los siguientes servicios:

- El envío de voz
- la posibilidad de realizar una llamada de emergencia a un número común (911) en forma gratuita y con cualquier compañía de que tenga cobertura.
- el servicio de mensajes cortos (SMS) enviados por canales independientes de señalización,
- Envío y recepción de fax digital.

Además de estos servicios existen los servicios portadores, dentro de los que se pueden incluir la transmisión de datos a 9600 bits por segundo (bps), aunque con el desarrollo que han sufrido los terminales en los últimos años esta tasa de velocidad puede alcanzar valores superiores a los 64Kbps mediante la agrupación de canales.

2.2.1 ARQUITECTURA

La arquitectura del sistema GSM está organizada en subsistemas, componentes e interfaces, el sistema tiene tres subsistemas: la estación móvil (Móvil Station), el subsistema de estación base (BSS-Base Station Subsystem) y el subsistema de conmutación (NSS-Network and Switching Subsystem) todos estos subsistemas intercambian datos mediante interfaces.

La estación móvil se comunica con el subsistema de estación base mediante el interfaz llamado radio y este con el subsistema de conmutación mediante el denominado interfaz. Desde este último sistema se intercambian los datos con la red telefónica fija.

La estación suele presentar el único elemento del sistema que el usuario llega a ver. Además de las funciones básicas de radio y de proceso necesarias para acceder a la red a través de la interfaz de radio, una estación móvil debe proporcionar un interfaz al usuario, mediante dispositivos de envío y recepción de datos.

[5] G. COLOURIS, J. DOLLIMORE, TIM, KINDBERG, SISTEMAS DISTRIBUIDOS, Conceptos y Diseño, Tercera edición, Editorial Addison Wesley 2001, pág. 79

La BSS está en contacto con las centrales de conmutación del NSS, la función del subsistema de estación base es la conexión entre las estaciones móviles y el subsistema de conmutación, resumiendo como la conexión entre usuarios de telecomunicaciones.

La interfaz de radio del sistema GSM es una combinación del Acceso Múltiple por División en Frecuencia (FDMA) y Acceso Múltiple por División de Tiempo (TDMA), la comunicación la realiza mediante la división del ancho de banda de cada usuario en distintos canales lógicos, que se utilizan para el transporte de información entre usuarios o la información de control del mismo sistema.

Según el tipo de información transportada se definen dos tipos de canales lógicos: el canal de control y los canales de tráfico, los canales de tráfico se utilizan para transportar la información entre usuarios, mientras que la función de los canales de control es transferir la información de la señalización, estos canales de control se pueden dividir en canales de control comunes y canales de control dedicados.

Los canales de control comunes llevan la información relativa a todos los móviles que se encuentren en la celda, convirtiéndose en un vínculo importante para el transporte de datos a los usuarios, por ejemplo de la existencia de una llamada, de la asignación de un canal para que otro no lo solicite, en la siguiente tabla encontramos los canales comunes.

CANALES DE CONTROL COMUN	UTILIZADOS POR LA ESTACION BASE
NOMBRE	CONOCIDO COMO
CANAL DE CONTROL DIFUSIVO	BCCH-BROADCAST CONTROL CHANEL
CANAL DE BÚSQUEDA	PCH-PAGING CHANEL
CANAL DE ACCESO ALEATORIO	RACH-RANDOM ACCESS CHANNEL
CANAL DE ACCESO GARANTIZADO	AGCH-ACCESS GRANT CHANNEL

TABLA 2.3 CANALES DE CONTROL COMÚN UTILIZADOS POR LA ESTACIÓN BASE

Los canales de control dedicados a un móvil llevan la información referente y orientada para un único Terminal, en la siguiente tabla especificamos los canales dedicados.

CANALES DE CONTROL DEDICADOS	UTILIZADOS POR LA ESTACION BASE
NOMBRE	CONOCIDO COMO
CANAL DE CONTROL DEDICADO	SDCCH-STAND-ALONE DEDICATED CONTROL
CANAL DE CONTROL ASOCIADO	SACCH-SLOW ASSOCIATED CONTROL
CANAL DE CONTROL ASOCIADO	FACCH-FAST ASSOCIATED CONTROL

TABLA 2.4 CANALES DE CONTROL DEDICADOS UTILIZADOS POR LA ESTACIÓN BASE

El Subsistema de Conmutación (NSS) incluye las funciones básicas de conmutación del GSM, así como las bases de datos para los usuarios y la gestión de movilidad, siendo su función principal el de gestionar las comunicaciones entre los usuarios GSM y los usuarios de otras redes de telecomunicación, como puede ser para otras redes de sistemas móviles en la Figura 2 se detalla la arquitectura del sistema GSM.

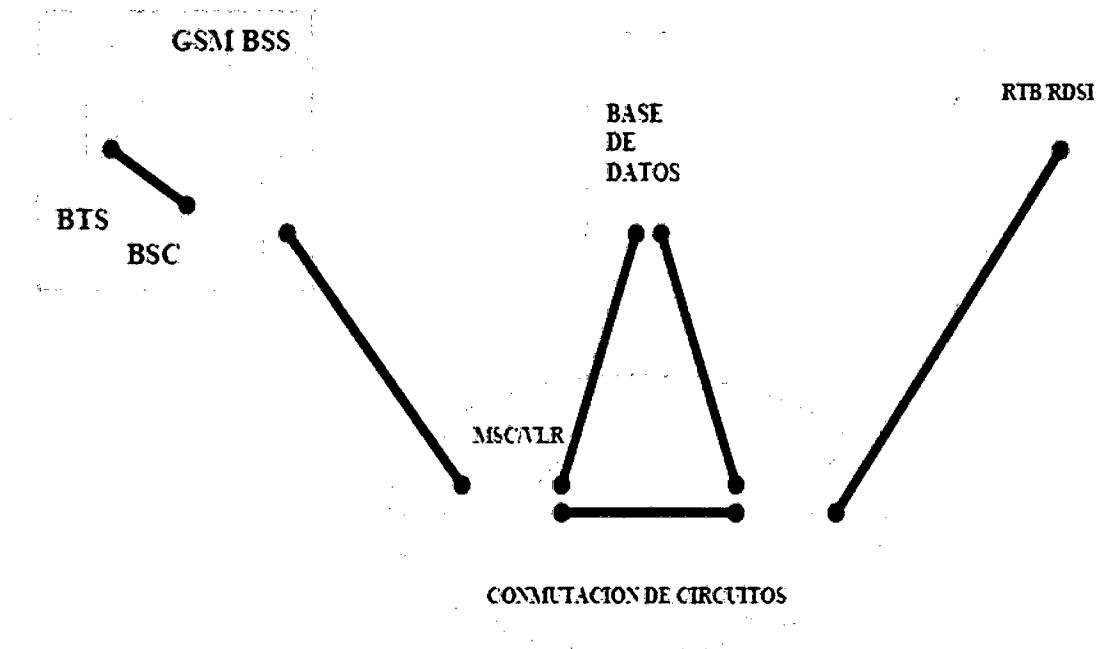


FIG. 2.6 ARQUITECTURA DEL SISTEMA GSM

Con los sistemas analógicos cualquier persona con una estación de radioaficionado o con un escáner de radio escuchaba las conversaciones del vecino, pero con el sistema GSM esta tarea se vuelve complicada y queda reservada a equipos sofisticados ya que en la conversación digital el escáner tiene que ser conectado a un ordenador que tendrá el trabajo de interpretar una señal que esta digitalizada, entrelazada y comprimida esta información por lo general está cifrada mediante claves y ciertos algoritmos de encriptación que en principio se creían indescifrables. En el sistema GSM existe un centro de autenticación que es el encargado de autorizar al móvil a introducirse a la red, en donde cada usuario tiene una clave secreta la cual es asignada en el momento de contratar el servicio y está localizada en dos sitios en la SIM card y en el móvil siendo el sistema el encargado de verificar las claves personales y de realizar la respectiva autenticación.

2.2.2 DESARROLLO DE UNA LLAMADA GSM

En la siguiente grafica se detalla un ejemplo de enrutamiento de llamadas para GSM.

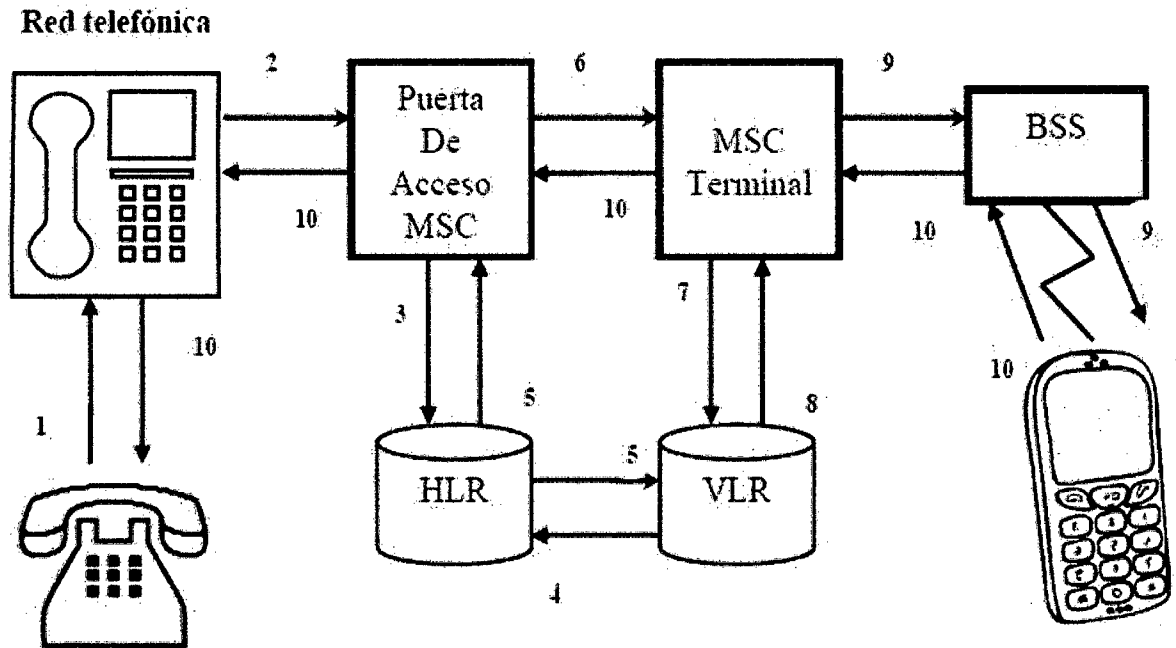


FIG. 2.7 EJEMPLO DE UN ENRUTAMIENTO DE LLAMADAS GSM

Paso 1: Un usuario de telefonía llama a la unidad móvil a través de la red pública.

Paso 2: La llamada se enruta a un MSC, el que se encarga de examinar los dígitos marcados y determina que no puede enrutar la llamadas más lejos.

Paso 3: El registro de ubicación HLR interroga al usuario llamado.

Paso 4: El HLR interroga el registro de ubicación del visitante VLR que actualmente está dando servicio al usuario.

Paso 5: El VLR devuelve un número de enrutamiento al HLR que lo devuelve al MSC.

Paso 6: Con este número de enrutamiento, el MSC enruta la llamada al MSC Terminal.

Paso 7: El MSC pide al VLR correlacionar la llamada con el subscritor.

Paso 8: El VLR realiza la acción que se le solicite.

Paso 9: La BSS recibe una solicitud de notificación del MSC Terminal y envía una señal de notificación.

Paso 10: Cuando la señal de usuario regresa, la llamada se completa.

2.2.3 VENTAJAS Y DESVENTAJAS ANTE OTRAS TECNOLOGÍAS

2.2.3.1 VENTAJAS

- GSM a nivel mundial tiene un mayor número de usuarios que CDMA.
- GSM cuenta con el mayor número de operadoras celulares a nivel mundial.
- GSM incorpora en su sistema GPRS una ventaja tecnológica para brindar servicios 3G.
- Con la implementación de la banda GSM de 850 Mhz permitió el crecimiento de esta tecnología en Latinoamérica.
- Roaming internacional.
- Tecnología de implementación relativamente económica en relación a CDMA lo cual permite obtener rentabilidad económica en poco tiempo.
- Uso eficiente del espectro radioeléctrico.
- Facilidad para la transmisión de datos inalámbricos.
- GSM brinda claridad de voz en las llamadas.
- GSM da la facilidad de cambiar de dispositivo móvil mediante el SIM.
- GSM puede operar en cuatro bandas 850 Mhz, 900 Mhz, 1800 Mhz, 1900 Mhz.
- GSM permite el envío y recepción de información multimedia.
- GSM tiene su arquitectura abierta lo cual brinda una compatibilidad con otras tecnologías.

2.2.3.2 DESVENTAJAS

- GSM a nivel tecnológico es inferior a CDMA.
- CDMA utiliza el espectro radioeléctrico de una manera más eficiente que GSM.
- GSM no posee el nivel de seguridad que tiene CDMA.
- En CDMA las comunicaciones son codificadas con lo cual se logra una mayor cantidad de enlaces.
- CDMA posee un sistema que le permite una mayor velocidad en la transmisión de datos más o menos unos 144 Kbps.
- CDMA no es propenso a interferencia externa.
- GSM requiere un número considerado de radio bases para brindar una buena cobertura.

2.2.4 LIMITACIONES DE GSM PARA LA TRANSMISIÓN DE DATOS

Las redes GSM tienen ciertas limitaciones para la transmisión de datos:

- Velocidad de transferencia de 9,6 Kbps.
- Tiempo de establecimiento de conexión, de 15 a 30 segundos.
- Además las aplicaciones deben ser reinicializadas en cada sesión.
- Pago por tiempo de conexión.
- Problemas para mantener la conectividad en itinerancia (Roaming).

2.2.5 MÓDULO GPRS/GSM (CELULAR) SIM900

El módulo GSM/GPRS se basa en el controlador SIM900 de SIMCOM y está diseñado para trabajar con Arduino y sus versiones compatibles. Este módulo permite comunicarte usando la red GSM de telefonía celular. Con este podrás acceder a los servicios SMS, MMS, GPRS y Telefonía de una manera sencilla enviando comandos AT. Asimismo tiene incorporadas en la placa 12 GPIOs, 2 PWM y un ADC propios del módulo SIM900.

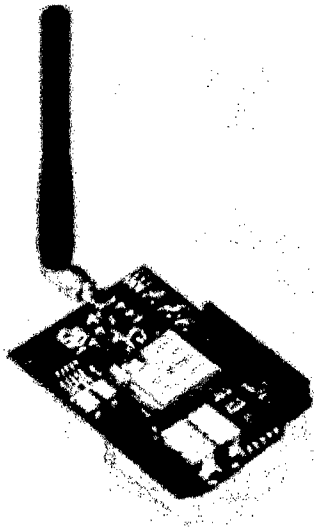


FIG. 2.8 MÓDULO GPRS/GSM SIM 900

2.2.5.1 ESPECIFICACIONES TÉCNICAS

Quad-Band 850 / 900/ 1800 / 1900 MHz - funciona en todas las redes celulares del planeta.

GPRS multi-slot class 10/8

Estación móvil GPRS clase B

Cumple con GSM phase 2/2+

Clase 4 (2 W @ 850 / 900 MHz)

Clase 1 (1 W @ 1800 / 1900MHz)

Controlable vía comandos AT estándar: GSM 07.07 & 07.05 Comandos mejorados: SIMCOM AT Commands.

Servicio corto de mensajes (SMS): para poder enviar pequeños paquetes de datos a través de la red celular.

Pila TCP/UDP incorporada: permite enviar datos a un servidor web.

Incorpora un reloj en tiempo real: RTC (Requiere pila)

Puerto serial configurable para comunicación con el módulo.

2.2.5.2 DIAGRAMA FISICO DEL HARDWARE Y DESCRIPCIÓN

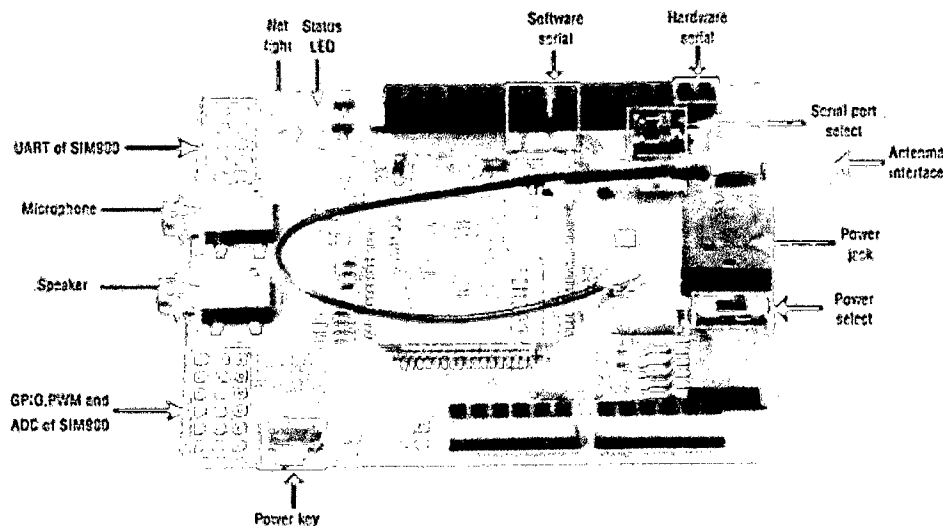


FIG. 2.9 DIAGRAMA FISICO DEL MODULO SIM 900

- Selección de alimentación: Selecciona el suministro eléctrico del módulo GPRS (La alimentación es de 5V)
- Jack de alimentación: Conexión externa de 4.8V a 5V DC de alimentación
- Interface de antena: Conexión con antena externa
- Selección de puerto serial: Selección de puerto serie a través de software y hardware que están conectados con el módulo GPRS
- Hardware Serial: D0/D1 de Arduino
- Software serial: D7/D8 de Arduino
- Estado de LED: Indica el encendido del modulo
- Net light: Indica la vinculación del módulo SIM900 con la red
- UART of SIM900 - UART pins breakout of SIM900
- Micrófono: Para responder la llamada telefónica
- Speaker: Para escuchar la llamada telefónica
- GPIO, PWM and ADC of SIM900 - GPIO, PWM and ADC pins breakout of SIM900
- Power key - power up and down for SIM900
- D0: D1: Sin usar se puede seleccionar el puerto para comunicación serie a través de software con el modulo
- D2: D3: D4: D5: D6: Sin usar
- D7: D8: Usar si selecciona el puerto serie para comunicación con el módulo
- D9: Usado para control por software the power up or down of the SIM900
- D10: D11: D12: D13: Sin usar
- D14 (A0): D15(A1): D16(A2): D17(A3): D18(A4): D19(A5): Sin usar

2.2.5.3 DIAGRAMA DE BLOQUES DEL MODULO SIM 900

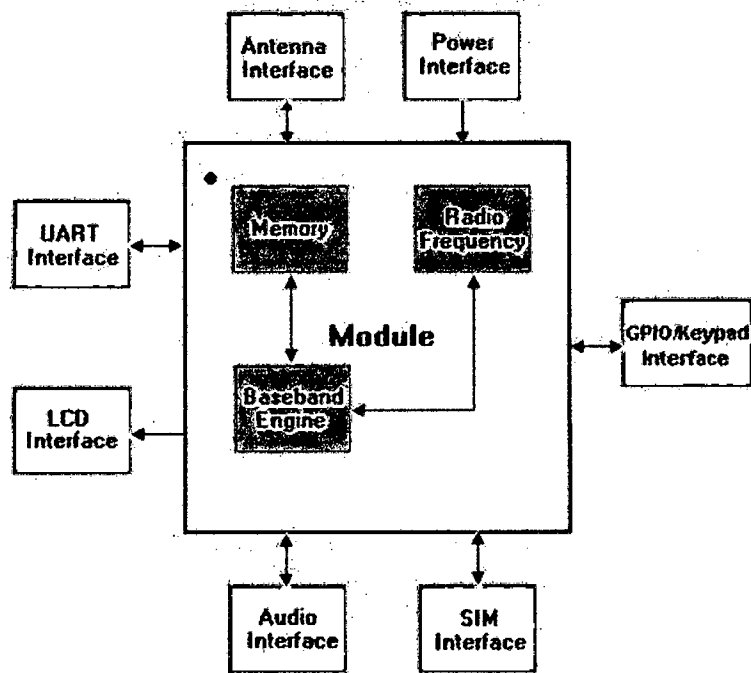


FIG. 2.10 DIAGRAMA LOGICO DEL SIM 900

La siguiente figura muestra un diagrama lógico del SIM900 e ilustra principalmente la parte funcional:

- El motor de banda base GSM
- El destello y SRAM
- La parte de radiofrecuencia GSM
- La interfaz de la antena
- Las otras interfaces

2.2.5.4 EVALUACIÓN DEL SIM900

Para ayudarle en la aplicación de SIM900, SIMCOM puede suministrar una Junta Directiva de Evaluación (EVB) que interconecta el SIM900 directamente con suministro de fuerza apropiado, agarradera de la tarjeta SIM, puerto RS232 serial portuario, del microteléfono del puerto, del auricular, línea en puerto, antena y todo GPIO del SIM900.

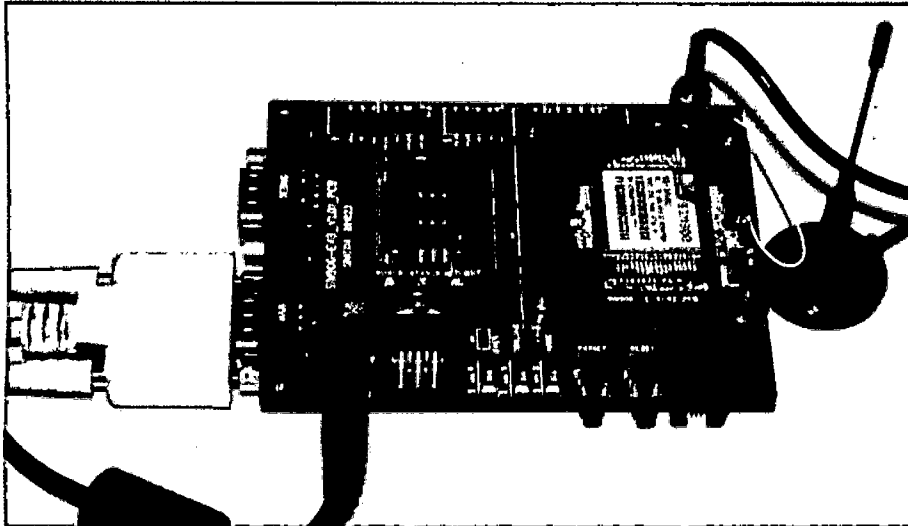


FIG. 2.11 FOTOGRAFIA DEL MODULO SIM 900

2.3 SISTEMA OPERATIVO ANDROID

2.3.1 ¿QUÉ ES ANDROID?

Android es un sistema operativo para dispositivos móviles como teléfonos inteligentes y tabletas basado en el núcleo Linux. Es desarrollado por la Open Handset Alliance, la cual es liderada por Google, usando diversos conjuntos de herramientas de software de código abierto para dispositivos móviles. Fue construido para permitir a los desarrolladores la creación de aplicaciones móviles que aprovechan al máximo el uso de todas las herramientas que un dispositivo como este puede ofrecer.

Implementa una arquitectura en la que cualquier aplicación puede obtener acceso a las capacidades del teléfono móvil. Por ejemplo, una aplicación puede llamar una o varias de las funcionalidades básicas de los dispositivos móviles, tales como realizar llamadas, enviar mensajes de texto, o utilizar la cámara, facilitando a los desarrolladores crear experiencias más ricas y con más coherencia para los usuarios.

Está construido sobre el kernel de Linux. Además, se utiliza una máquina personalizada virtual que fue diseñada para optimizar los recursos de memoria y de hardware en un entorno móvil. Android es de código abierto, y además puede ser libremente ampliado para incorporar nuevas tecnologías de vanguardia que van surgiendo. La plataforma continuará evolucionando a medida que la comunidad de desarrolladores trabajando juntos puedan crear aplicaciones móviles innovadoras.

2.3.2 HISTORIA DE ANDROID

Fue desarrollado por Android Inc., empresa que en 2005 fue comprada por Google, aunque no fue hasta 2008 cuando se popularizó, gracias a la unión al proyecto de Open Handset Alliance, un consorcio formado por 48 empresas de desarrollo hardware, software y telecomunicaciones, que decidieron promocionar el software libre. Pero ha sido Google quien ha publicado la mayor parte del código fuente del sistema operativo, gracias al software Apache, que es una fundación que da soporte a proyectos software de código abierto.

Dado que Android está basado en el núcleo de Linux, tiene acceso a sus recursos, pudiendo gestionarlo, gracias a que se encuentra en una capa por encima del Kernel, accediendo así a recursos como los controladores de pantalla, cámara, memoria flash.

2.3.3 VERSIONES DISPONIBLES

El sistema operativo Android, al igual que los propios teléfonos móviles, ha evolucionado rápidamente, acumulando una gran cantidad de versiones, desde la 1.0 para el QWERTY HTC G1, hasta la 4.4.2 que acaba de salir al mercado.

- CUPCAKE: ANDROID VERSIÓN 1.5

Características: Widgets, teclado QWERTY virtual, copy & paste, captura de vídeos y poder subirlos a Youtube directamente.

- DONUT: ANDROID VERSIÓN 1.6

Características: Añade a la anterior la mejora de la interfaz de la cámara, búsqueda por voz, y navegación en Google Maps.

- ECLAIR: ANDROID VERSIÓN 2.0/2.1

Características: Mejoras en Google Maps, salvapantallas animado, incluye zoom digital para la cámara, y un nuevo navegador de internet.

- FROYO: ANDROID VERSIÓN 2.2

Características: Incluye hotspot Wifi, mejora de la memoria, más veloz, Microsoft Exchange y video-llamada.

- GINGER BREAD: ANDROID VERSION 2.3

Características: Mejoras del consumo de batería, el soporte de vídeo online y el teclado virtual, e incluye soporte para pagos mediante NFC.

- HONEY COMB: ANDROID VERSION 3.0/3.4

Características: Mejoras para tablets, soporte Flash y Divx, integra Dolphin, multitarea pudiendo cambiar de aplicación dejando las demás en espera en una columna, widgets y homepage personalizable.

- ICE CREAM SANDWICH: ANDROID VERSION 4.0

Características: Multiplataforma (tablets, teléfonos móviles y netbooks), barras de estado, pantalla principal con soporte para 3D, widgets redimensionables, soporte usb para teclados, reconocimiento facial y controles para PS3.

2.3.4 ARQUITECTURA DE LA PLATAFORMA ANDROID

La arquitectura interna de la plataforma Android, está básicamente formada por 4 componentes: aplicaciones, almacén de aplicaciones, librerías y kernel/Linux. En la Figura 2.18, se muestran las capas que conforman el sistema operativo Android.

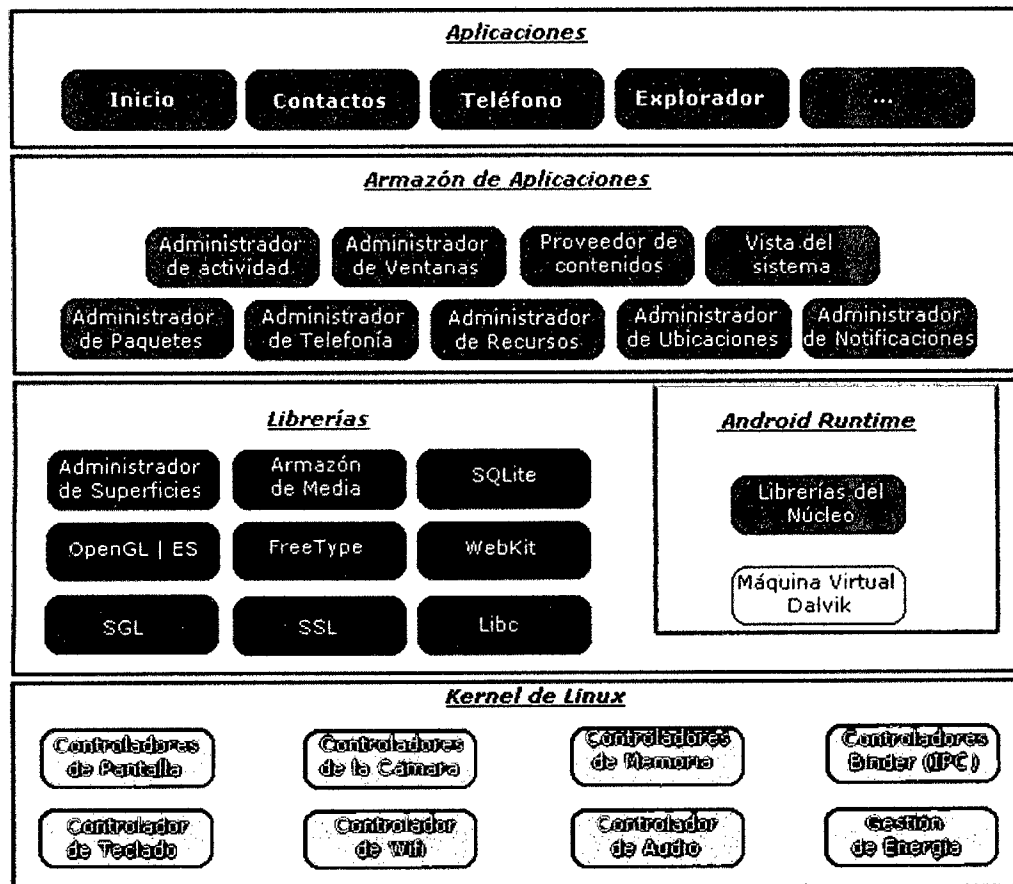


FIG. 2.12 SISTEMAS DE CAPAS DE ANDROID

2.3.4.1 KERNEL

Un núcleo o kernel es un software que constituye la parte más importante del sistema operativo. Es el principal responsable de facilitar a los distintos programas acceso seguro al hardware de la computadora o en forma básica, es el encargado de gestionar recursos, a través de servicios de llamada al sistema.

Como hay muchos programas y el acceso al hardware es limitado, también se encarga de decidir qué programa podrá hacer uso de un dispositivo de hardware y durante cuánto tiempo, lo que se conoce como multiplexado. Acceder al hardware directamente puede ser realmente complejo, por lo que los núcleos suelen implementar una serie de abstracciones del hardware. Esto permite esconder la complejidad, y proporciona una interfaz limpia y uniforme al hardware subyacente, lo que facilita su uso al programador.

El kernel lo podemos definir como el corazón del SO. En esta serie los modelos de desarrollo han cambiado, la manera de numerarse es de 4 dígitos (VV.RR.NR.CR).

VV: Indica la versión (o serie) del kernel.

RR: Indica la revisión del kernel (Da igual que los vea de forma impar o par, hoy en día, no tiene significado)

NR: Indica nuevas revisiones del kernel. Estos números cambian cuando se incorporan nuevas características y drivers

CR: Este dígito cambia cuando se corrigen fallos de programación o fallos de seguridad dentro de una revisión.

Android utiliza el núcleo de Linux 2.6 como una capa de abstracción para el hardware disponible en los dispositivos móviles. Esta capa contiene los drivers necesarios para que cualquier componente hardware pueda ser utilizado mediante las llamadas correspondientes. Siempre que un fabricante incluye un nuevo elemento de hardware, lo primero que se debe realizar para que pueda ser utilizado desde Android es crear las librerías de control o drivers necesarios dentro de este kernel de Linux embebido en el propio Android.

2.3.5 GUIA PARA DESARROLLADORES (BÁSICA)

La Guía ofrece una introducción práctica a desarrollar aplicaciones para Android y documentación sobre las características de las plataformas más importantes. Se estudian los conceptos detrás de Android, el marco para la construcción de una aplicación y las herramientas para el desarrollo, la prueba y la publicación de software para la plataforma.

La Guía del desarrollador tiene la mayor parte de la documentación para la plataforma Android, con excepción de material de referencia en el Framework de la API, para conocer las especificaciones de la API.

ANDROID BASICS

Es una orientación inicial para Android, lo que es, lo que ofrece y cómo se ajusta su aplicación.

TEMAS DEL FRAMEWORK

Son las discusiones sobre determinadas partes del framework de Android y la API. Para una introducción al marco, comienzan con Application Fundamentals; luego explora otros temas, desde el diseño de una interfaz de usuario y la creación de recursos para el almacenamiento de datos; y el uso de los permisos.

TEMAS DEL ANDROID MARKET

La documentación de los temas que conciernen a la publicación y monetización de las aplicaciones en AndroidMarket, por ejemplo, cómo hacer cumplir las políticas de concesión de licencias y ponerlo en práctica en la aplicación de facturación.

DESARROLLO

Indicación es para el uso de desarrollo de Android y herramientas de depuración, y para comprobar los resultados.

PUBLICACIÓN

Las instrucciones sobre cómo preparar su aplicación para la implementación y la forma de publicar cuando esté listo.

APLICACIONES WEB

Hace referencia a la documentación sobre cómo crear aplicaciones web que funcionan perfectamente en dispositivos con Android y como crear aplicaciones Android para incrustar contenido basado en web.

APÉNDICE

Información de referencia y especificaciones, así como preguntas frecuentes, un glosario de términos, y otra información. El primer paso en la programación para Android es la descarga del SDK (software development kit). Después de tener el SDK, se empieza por buscar a través de the Dev Guide. Si se quiere empezar por conseguir un rápido vistazo a algo de código, el tutorial Hola Mundo los lleva a través de una aplicación "Hello World" para introducir algunos conceptos básicos de una aplicación Android.

2.3.6 PAUTAS PARA LAS INTERFACES DE USUARIO

El sistema operativo Android fue adoptado con gran rapidez por múltiples fabricantes de dispositivos móviles, que lo adoptaron como plataforma debido a su carácter más abierto.

Esto potenció a Android, y su tienda de aplicaciones (el AndroidMarket) comenzó a contar con una cantidad enorme de desarrollos (a la fecha, más de 400.000). Sin embargo, esto mismo generó en Android una falta de estandarización en las diferentes aplicaciones, que ha generado también ligeras incompatibilidades entre las distintas implementaciones del sistema operativo de los fabricantes.

Para la última versión mayor de Android (4.0) Google comenzó a tomar algunas cartas en el asunto, y para ello liberó una página llamada “AndroidDesign” que define una serie de principios para el diseño de interfaces y afirman:

“Estos principios de diseño fueron desarrollados por y para el Equipo de Experiencia de Usuario de Android teniendo en mente las mejores intenciones y consideraciones. Se deben tener en cuenta a la hora de aplicar las ideas creativas de diseño.” [9]

versión 4.0 de Android, pocas aplicaciones del AndroidMarket las cumplen, y en este momento no son un requerimiento para conseguir la aprobación en la tienda oficial de Android. La página de “AndroidDesing” inicia planteando una serie de principios generales para el diseño de interfaces gráficas.

Las aplicaciones nativas de Android siguen tres principios generales:

Encántame: Las aplicaciones deben combinar belleza, simplicidad y propósito, para crear experiencias poderosas y de mínima dificultad de uso.

Simplifica mi vida: Las aplicaciones de Android deben facilitar la vida y ser fáciles de entender. Cuando las personas utilicen por primera vez una aplicación, deben deducir de manera intuitiva las características más importantes.

Sorpréndeme: No es suficiente el hacer una aplicación fácil de utilizar. Las aplicaciones de Android deben empoderar a las personas a intentar nuevas cosas y a usarlas de manera creativa.

Todos estos principios generales de diseño se traducen en una serie de elementos más concretos agrupados en tres grandes grupos: Estilo, patrones y bloques de construcción.

[9] Android Developers. Design. [En línea].
<http://developer.android.com/guide/index.html>. [Citado el 16 de agosto del 2014]

2.3.7 APP INVENTOR

App Inventor es una aplicación originalmente desarrollada por Google y mantenida ahora por el Instituto de Tecnología de Massachusetts. Permite que cualquier persona, incluyendo las no familiarizadas con la programación y SDK de Android, pueda crear aplicaciones de Software para Android. Utiliza una interfaz gráfica, muy similar al Scratch y el StarLogo, que permite a los usuarios arrastrar y soltar objetos visuales para crear una aplicación que puede ejecutarse en el sistema Android.

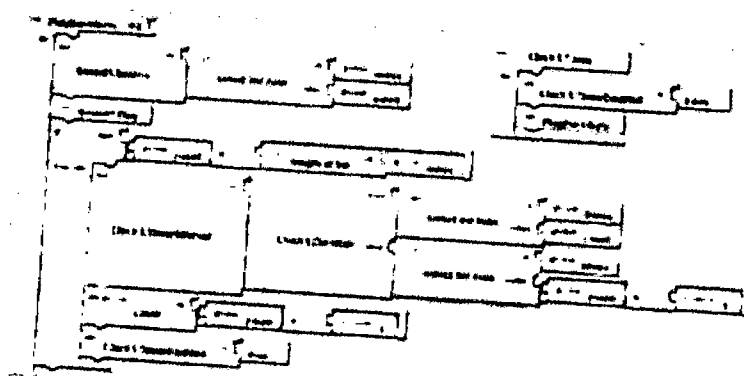


FIG. 2.13 ENTORNO DE DESARROLLO DE APP INVENTOR

Google puso fin al desarrollo el 31 de diciembre de 2011 cediéndole el código al MIT, quién lo ha puesto a disposición de todos. Se trata de una utilidad Web desarrollada por Google que permite realizar aplicaciones para Android sin escribir código Java, todo de forma visual e intuitiva (uniendo piezas de un puzzle).

Una característica interesante es que el desarrollo de la aplicación es en Web. Aunque es necesario instalar un módulo de software en la computadora, en el momento del desarrollo se ejecuta la última versión del App Inventor disponible en su sitio web y los proyectos se guardan en línea. El App Inventor consta de dos segmentos principales:

- Un módulo Web y
- El editor de bloques de Android.

El módulo en Web que se mencionaba, donde aparte de ser el punto de entrada tenemos acceso a nuestros proyectos y, una vez abierto un proyecto, podemos entrar a la sección de diseño de nuestra aplicación. Esta sección es donde podemos añadir los componentes y configurarlos apropiadamente. Si se trata de componentes visuales, entonces definimos también el diseño de la interfaz. Para los familiarizados con desarrollo de aplicaciones mediante componentes visuales verán que es un concepto bastante similar.

El segmento del editor de bloques se verá más adelante, por el momento basta con mencionar que ahí es donde los bloques se conectan cual piezas Lego, formando la lógica de la aplicación.

Permite a cualquiera crear aplicaciones de software para el sistema operativo Android.

Utiliza una interfaz gráfica que permite a los usuarios arrastrar y soltar objetos visuales para crear una aplicación que puede ejecutarse en el sistema Android, que funciona en muchos dispositivos móviles.

Todo ello sin usar ni una sola línea de código, de forma intuitiva y gráfica. La aplicación se puso a disposición de los usuarios, mediante invitación, el 12 de julio de 2010, el 15 de diciembre de 2010 se puso a disposición de usuarios registrados. La aplicación está dirigida a personas que no están familiarizadas con la programación de computadoras. La idea es que cualquier persona pueda desarrollarse sus propias aplicaciones para su dispositivo Android.

2.3.7.1 ¿PORQUÉ APP INVENTOR?

Porque es gratuito y poco exigente con los requisitos técnicos, funciona online y sin apenas instalación. Ha sido probado por estudiantes con excelentes resultados en institutos de enseñanza de San Francisco, eso es punto de partida interesante.

El lenguaje de programación que se usa en App Inventor es bastante similar al de Scratch, también desarrollado en el MIT. Ya hemos trabajado un poco con Scratch durante este curso. Cuando conocimos App Inventor no tuvimos ninguna duda. Había que probarlo.

2.3.8 REQUERIMIENTOS DEL SISTEMA

COMPUTADORA Y SISTEMA OPERATIVO

- **Macintosh** (with Intel processor): Mac OS X 10.5, 10.6.
- **Windows**: Windows XP, Windows Vista, Windows 7.
- **GNU/Linux**: Ubuntu 8+, Debian 5+.

NAVEGADOR WEB

- **Mozilla Firefox 3.6** o superior.
- **Apple Safari 5.0** o superior.
- **Google Chrome 4.0** o superior.
- **Microsoft Internet Explorer 7** o superior.

Para poder acceder al App Inventor se debe haber instalado Java 6 o superior en el ordenador o PC ya que no funcionará en el equipo, lo recomendable es no tratar de utilizar la aplicación App Inventor sin haberlo instalado.

Luego de ello se debe crear una cuenta en Gmail para poder tener un usuario y clave para acceder al App Inventor. La interfaz de esta herramienta se muestra a continuación:

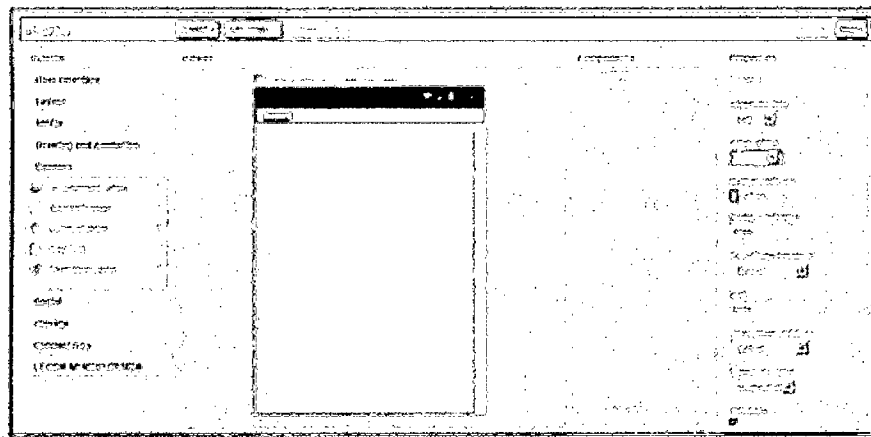






FIG. 2.14 INTERFAZ DE APP INVENTOR 2

Para realizar pruebas de la aplicación creada tenemos:

- 

 Si se está utilizando un dispositivo Android y se tiene un cable USB, se puede enviar la aplicación al dispositivo Android e instalarlo.
- 

 Se puede hacer uso de un emulador.

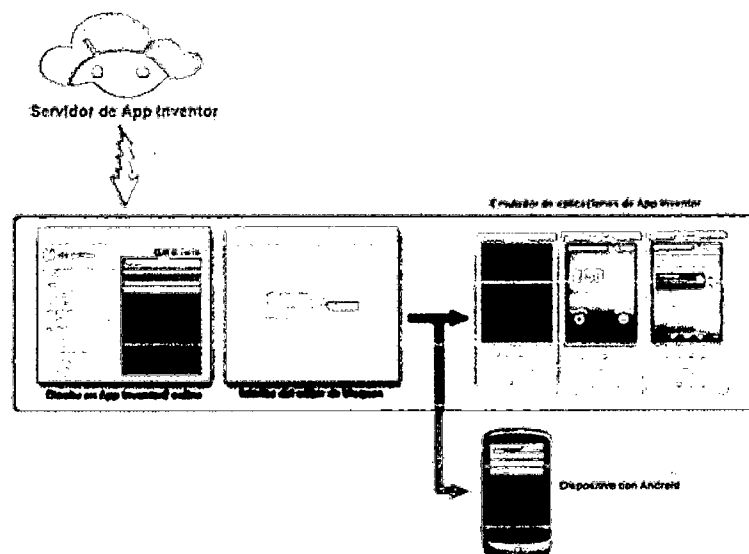


FIG. 2.15 USO DE EMULADOR VIRTUAL PARA PRUEBA DE LA APLICACIÓN

3.0 DISEÑO DEL HARDWARE DEL SISTEMA MONITOREO Y AHORRO DE ENERGÍA PARA REFEER.

3.1 DIAGRAMA DE BLOQUES DEL SISTEMA DE MONITOREO GSM/GPS Y AHORRO DE ENERGÍA PARA CONTENEDORES DE PRODUCTOS CONGELADOS REFEER

El diagrama de bloques del sistema que se desea es el que se muestra en la Figura 3.1

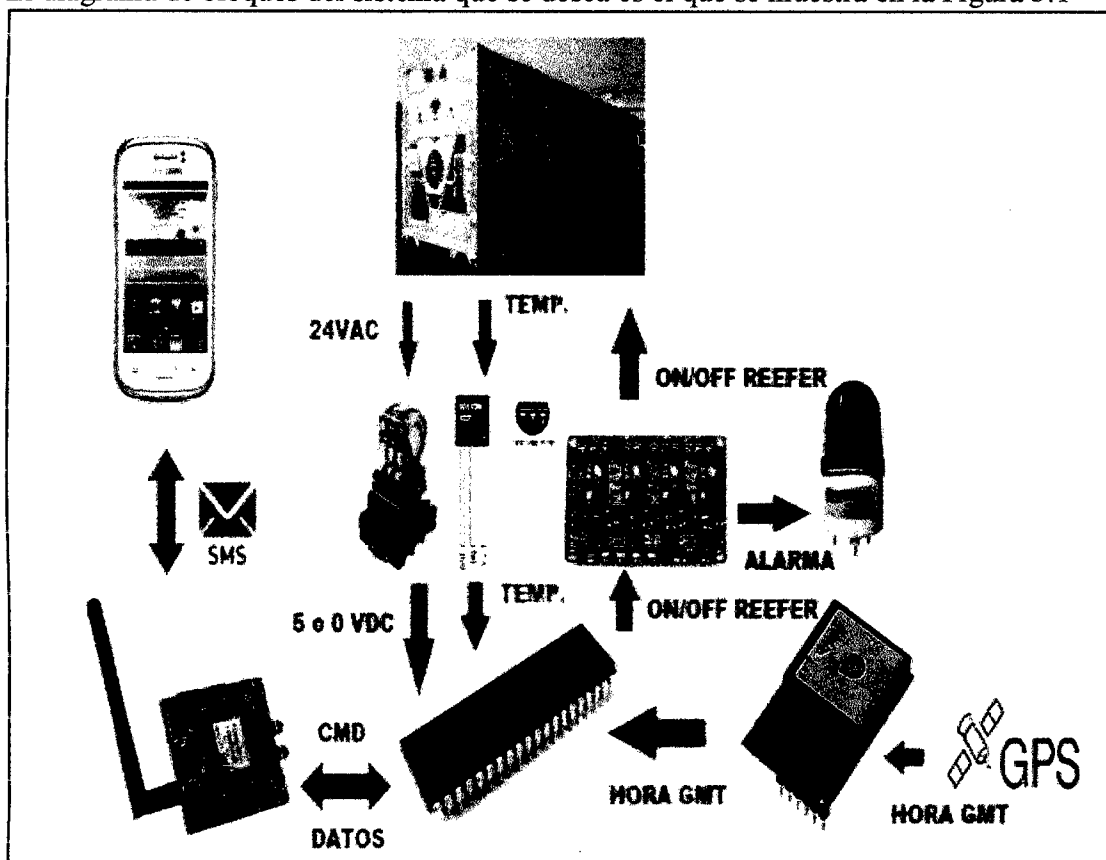


Figura 3.1 Sistema de monitoreo y ahorro de energía GSM/GPS

El diagrama de bloques muestra que se está monitoreando la temperatura mediante un sensor digital DS18B20 la cual si pasa de ciertos límites previamente programados enviara un mensaje de alerta , tambien se verifica que el reefer si esta con energia electrica para lo cual se utiliza un relay que se activa con 24VAC en bobina que proviene del sistema de control del reefer, este relay tiene dos salidas una con 5VDC y 0VDC el cual indicaria al microcontrolador si esta con energia o no, sino hay energia se envia un mensaje de texto inidcando que un reefer no tiene energia electrica.

El sistema cuenta con GPS para que el microcontrolador pueda leer la Hora GMT, lo cual permitira programar la hora en que el contenedor sele cortara la energia y la hora en la que conectara nuevamente la energia.

3.2 DISEÑO DE LA TARJETA ELECTRÓNICA PARA SISTEMA DE MONITOREO GSM/GPS

En la Figura 3.2 se muestra la tarjeta que se encargara de la detección de falta de energía y con conexión con modem GSM para el envío de alertas.

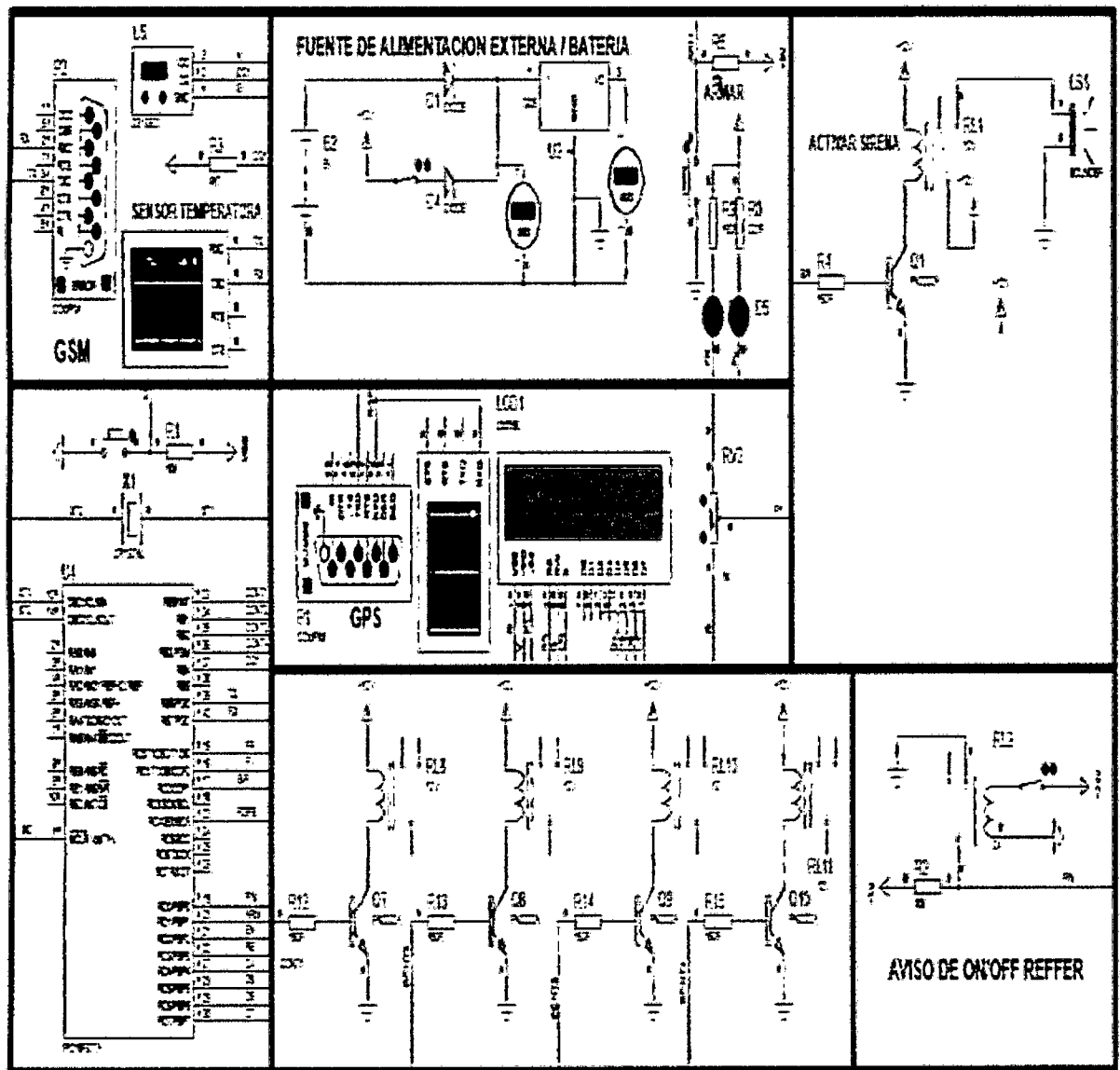


Figura 3.2 Tarjeta que se encargara del monitoreo de temperatura y energia electrica GSM/GPS.

Se procede a describir los componenets importantes de la tarjeta electrónica propuesta.

3.2.1 MICROCONTROLADOR PIC16F877A

Para la implementación de la tarjeta Electrónica se utiliza el microcontrolador PIC16F877A de Microchip que cumple con requerimientos de Puertos de E/S para el sistema que se propone así como la cantidad de memoria de Programa y Datos para el código de programa.

Los puertos de este microcontrolador se conectara como sigue: (Figura 3.3)

- Leer sensor digital DS18B20

- Conectar 1 relé de 24VAC en Bobina, con salida de 0 ó 5VDC.
- Conectar 4 relés para ON/OFF contenedor
- Conectar relé para sirena
- Conectar LCD de 16x2
- Comunicación RS232 con Modem GSM.SIM900.
- Comunicación con GPS

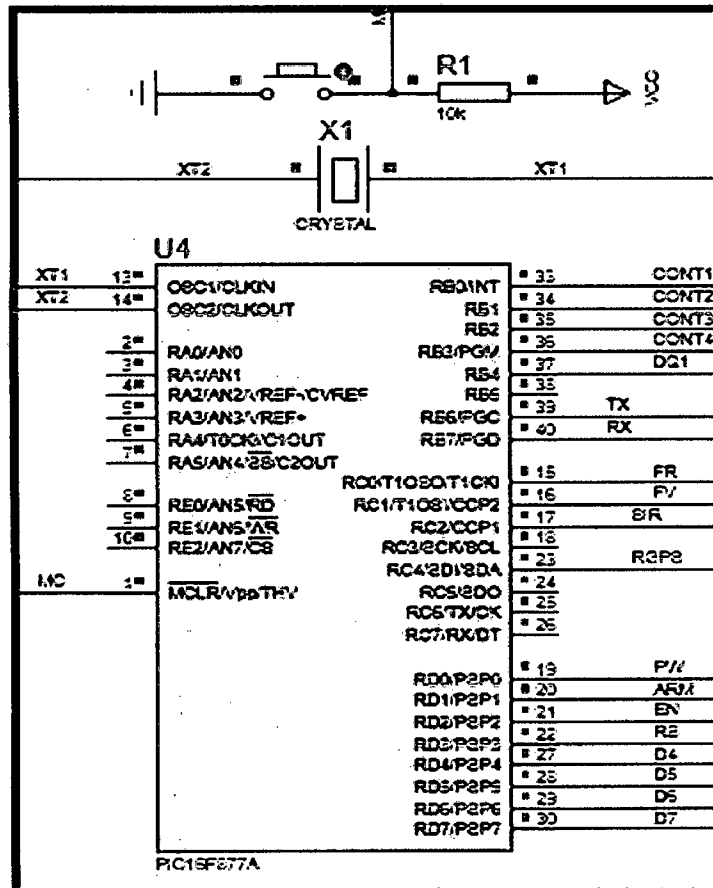


Figura 3.3. Microcontrolador PIC16F877A conexiones

3.2.2 CIRCUITO PARA ON/OFF DEL CONTENEDOR.

En la Figura 3.4 se muestra el circuito el cual se conecta al interruptor de 4 contactos del contenedor el cual permitirá apagar o encender el contenedor de manera automática o programada en hora y minutos según se programe por el usuario.

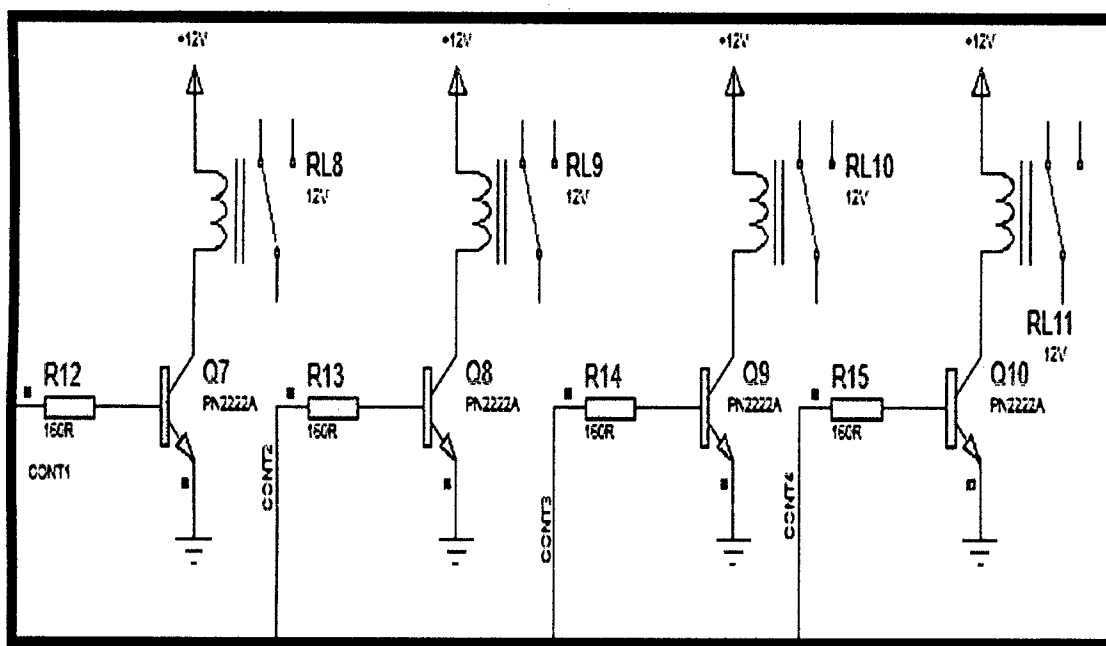


Figura 3.4 Circuito ON/OFF del contenedor

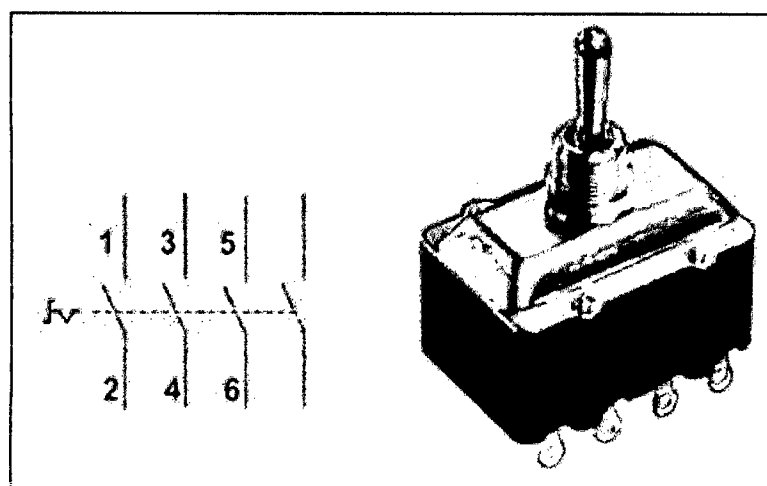


Figura 3.5 Interruptor de 4 contactos

3.2.3 VERIFICACIÓN DE ESTADO DEL CONTENEDOR ON-OFF

Se utiliza 1 refé de 11 pines como el que se muestra en la Figura 3.6

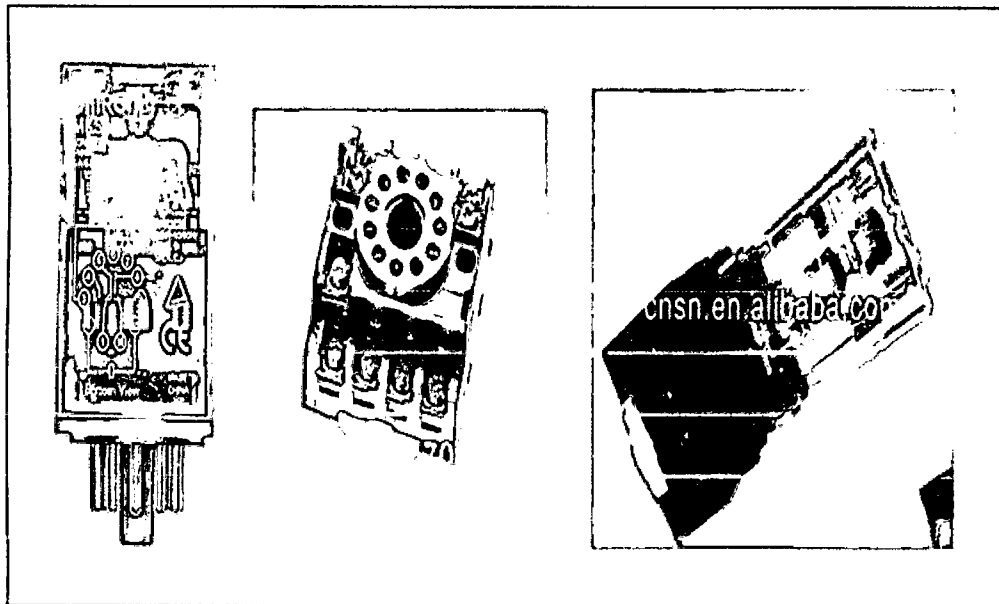


Figura 3.6 Relé de 11 Pines

El circuito de conexión por relé es el que se muestra en la Figura 3.7

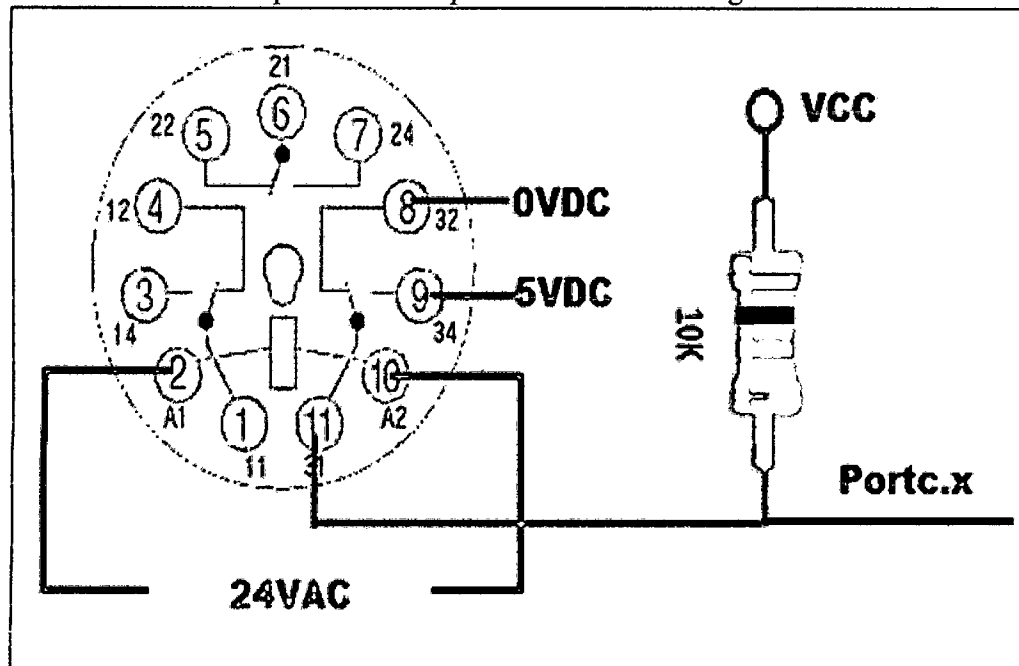


Figura 3.7 Circuito de conexión por relé de 11 pines

El circuito de conexión del relé es el que se muestra en la Figura 3.8.

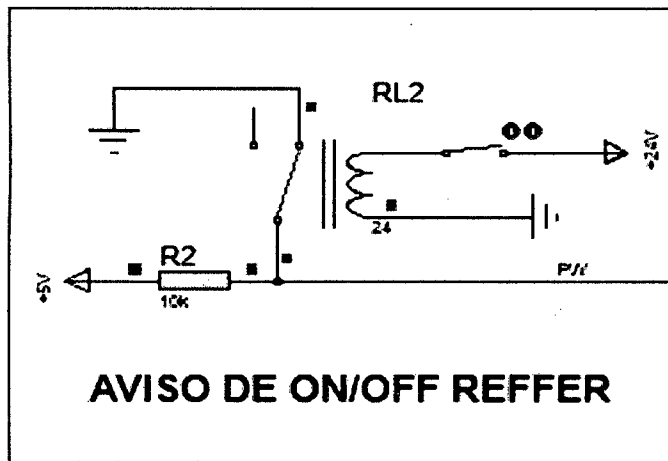


Figura 3.8 Conexión del relé para monitoreo de falta de energía

3.2.4 VISUALIZACIÓN LOCAL DE ESTADO DEL CONTENEDOR

Para la visualización del estado del contenedor se utiliza un LCD de 16x2 conectado al PORTD del microcontrolador como se muestra en la Figura 3.9. En este LCD se visualizara la temperatura actual, valor de referencia de temperatura, si hay energía eléctrica entre otros datos de importancia.

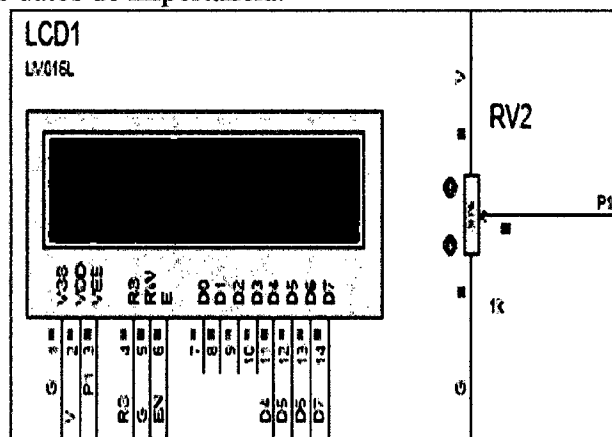


Figura 3.9 Visualización de estado de contenedores

3.2.5 CIRCUITO DE CONEXIÓN DEL GPS

En la Figura 3.10 se muestra como se conecta el GPS con el microcontrolador, solo se utiliza el pin de transmisión del GPS

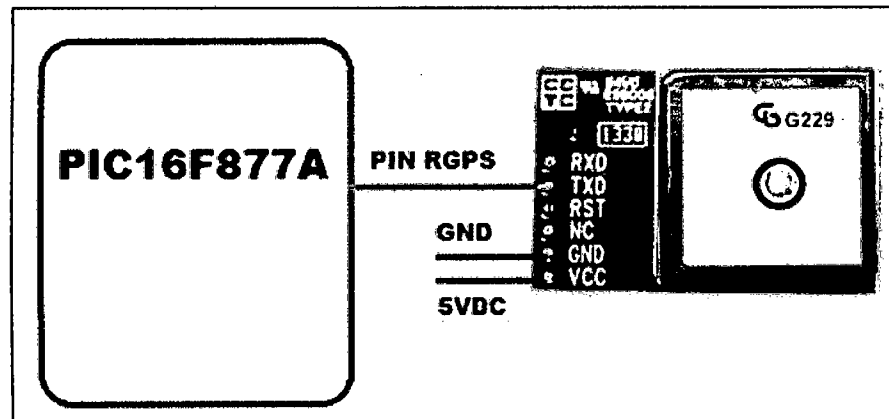


Figura 3.10. Conexión del GPS con microcontrolador

El GPS transmite una trama de datos como el que se muestra en la Figura 3.11, para el proyecto solo nos interesa el tiempo o UTC TIME del comando GGA o GPRMC

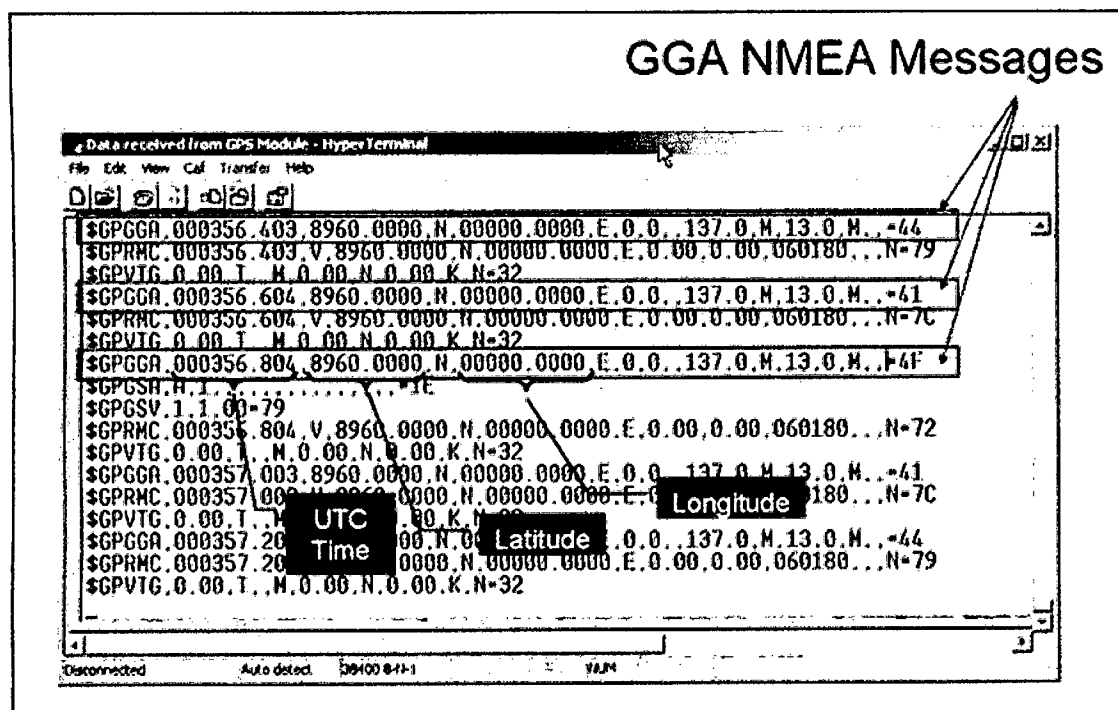


Figura 3.11. Trama de datos transmitido por el GPS

3.2.6 CIRCUITO DE CONEXIÓN CON EL SENSOR DE TEMPERATURA DS18B20

En las Figuras 3.12 y 3.13 se muestra como se conecta el sensor DS18B20 con el microcontrolador.

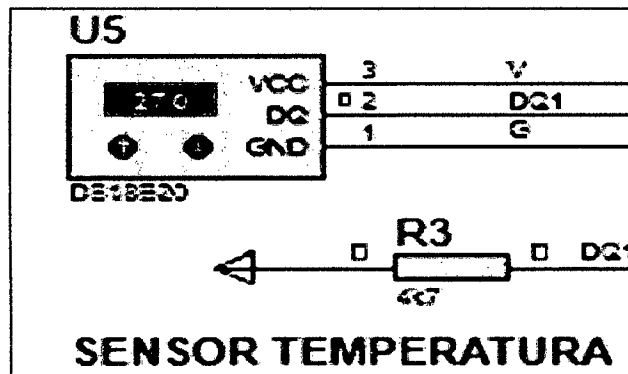


Figura 3.12. Conexión del sensor de temperatura DS18B20

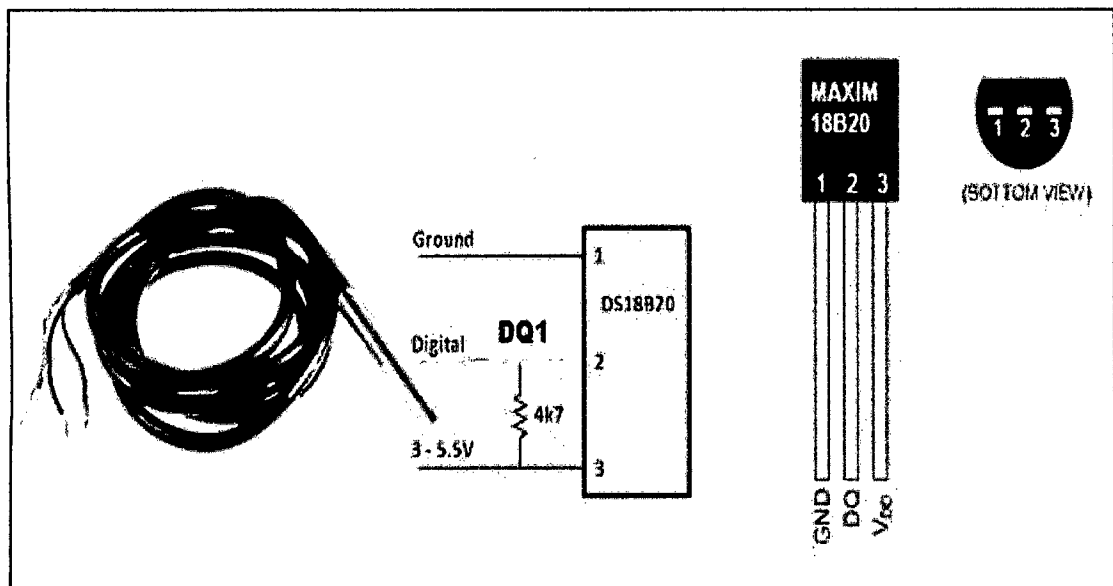


Figura 3.13. Sensor de temperatura DS18B20

3.2.7 ACTIVACIÓN DE ALARMAS

Para la activación de alarma/sirena se utiliza el relé conectado como se muestra en la Figura 3.14

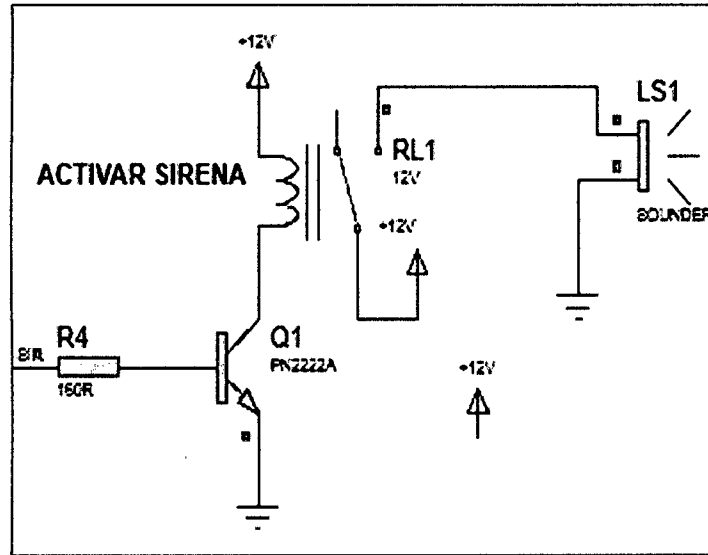


Figura 3.14 Circuito de activación de Alarma/Sirena

3.2.8 FUENTE DE ALIMENTACIÓN

La fuente de alimentación además de contar con una fuente de 12VDC también cuenta con una batería externa para fallos de energía eléctrica. (Figura 3.15).

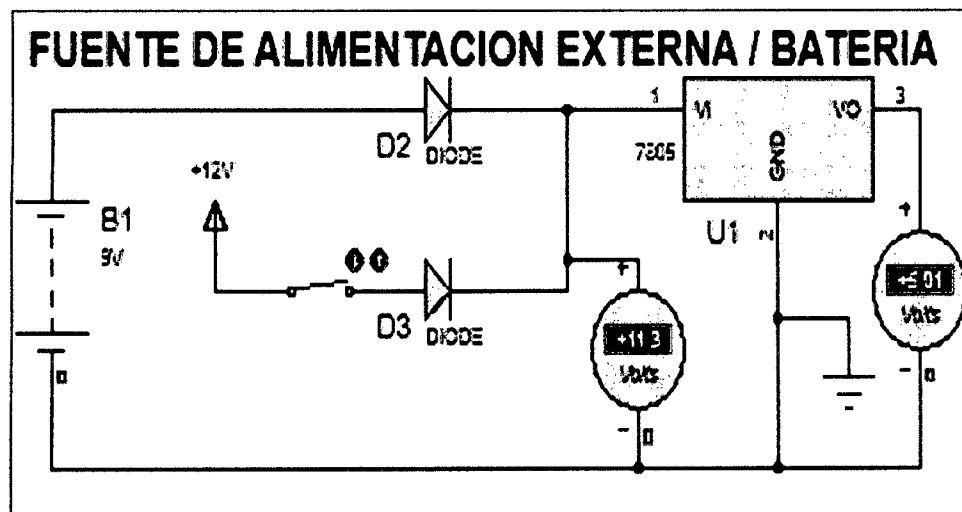


Figura 3.15 Fuente de alimentación de la tarjeta electrónica del sistema

3.2.9 COMUNICACIÓN CON EL MODEM GSM SIM900

Para el envío de comandos y mensajes de texto de alerta se usa el Modem GSM SIM900 (Figura 3.16) . En la Tabla 3.1 se muestra los comandos por SMS.

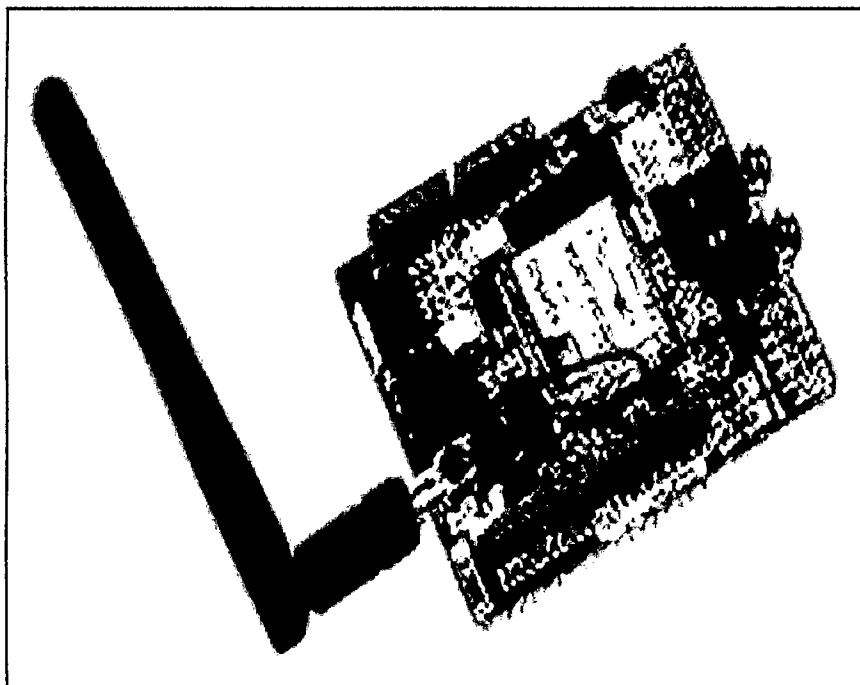


Figura 3.16 Modem GSM SIM900

Las conexiones del modem se muestra en la Figura 3.12

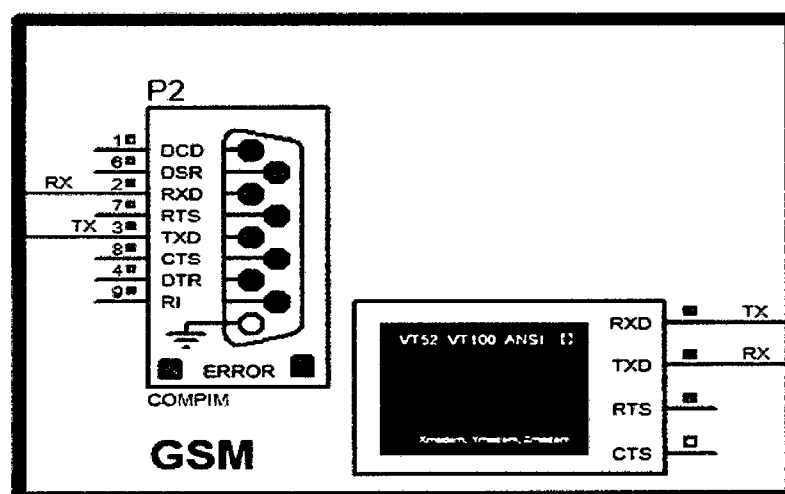


Figura 3.17. Conexiones del Modem GSM

Cuando se produce una alerta por falta de energía o temperatura se envía un mensaje de texto de alerta.

TABLA 3.1 TABLA DE COMANDOS POR SMS

	COMANDO	EJEMPLO	DESCRIPCION
1	RING		LLAMAR- RECIBE SMS CON VALOR DE LA TEMPERATURA Y SI CONTENEDOR ESTA ON/OFF
2	RT1NNNNNNNNNN	RT1951551591	PROGRAMAR CELULAR 1 PARA RECEPCION DE ESTADO Y ALERTAS : 951551591
3	RT2NNNNNNNNNN	RT2998822138	PROGRAMAR CELULAR 2 PARA RECEPCION DE ALERTAS : 998822138
4	RT3NNNNNNNNNN	RT3971231234	PROGRAMAR CELULAR 3 PARA RECEPCION DE ALERTAS : 971231234
5	RD1234123456	RD1234123456	RECIBE SMS CON LOS NUMEROS DE CELULARES PROGRAMADOS
6	RA1234123456	RA1234123456	SE PROGRAMA PARA QUE SISTEMA ENVIE UN SMS DE ALERTA
7	RB1234123456	RB1234123456	SE PROGRAMA PARA QUE SISTEMA ENVIE TRES SMS DE ALERTA
8	RE1234123456	RE1234123456	SE PROGRAMA PARA QUE SISTEMA ENVIE CONTINUAMENTE SMS DE ALERTA
9	RCXXXXX123456	RC1234123456	SE PROGRAMA PARA QUE SISTEMA NO ENVIE SMS
10	RWH112341234	RWH112341234	SE ACTIVA LOS RELES DE ENCENDIDO DEL CONTENEDOR
11	RWL112341234	RWL112341234	SE DESACTIVA LOS RELES DEL CONTENEDOR
12	RWZHHMMHhmm l	RWZ102511501	SE PROGRAMA PARA QUE EL CONTENDOR SE APAGE A LAS 10:25 Y SE ENCIENDA A LAS 11:50
13	RSVALORA5678912	RS-15A5678912	SE PROGRAMA LA REFERENCIA DE TEMEPRATURA A -15 °C
14	RY1234123456	RY1234123456	SE PROGRAMA EL ENCENDIDO Y APAGADO DE CONTENEDOR EN MODO AUTOMATICO
15	RY1234223456	RY1234223456	SE PROGRAMA EL ENCENDIDO Y APAGADO DE CONTENEDOR EN MODO MANUAL

3.3 DESCRIPCIÓN DEL CÓDIGO DE PROGRAMA DEL MICROCONTROLADOR

En esta parte del capítulo se procederá a describir brevemente las diferentes partes del código del programa del microcontrolador. Este código ha sido desarrollado en PICBASIC PROCOMPILER.

En la Figura 3.18 se muestran las definiciones necesarias para utilizar el LCD de 16X2 y configurar los puertos analógicos como puertos Digitales, entre otras variables que se utilizarán en el programa

```
DEFINE OSC 4
'DEFINICIONES PARA LCD
DEFINE LCD_DREG      PORTD
DEFINE LCD_DBIT      4
DEFINE LCD_RSREG     PORTD
DEFINE LCD_RSBIT     3
DEFINE LCD_EREG      PORTD
DEFINE LCD_EBIT      2
'*****
ADCON1 = %00000110 ! CONFIGURAR PORTA COMO I/O DIGITALES
'*****
'SALIDAS TIPO RELAY
CONTACTO1 VAR PORTB.0:CONTACTO2 VAR PORTB.1: CONTACTO3 VAR PORTB.2
CONTACTO4 VAR PORTB.3:SIRENA VAR PORTC.2:FOCO_ROJO VAR PORTC.0
FOCO_VERDE VAR PORTC.1:TRISD.0=1:TRISB.4=1
TRISB.0=0:TRISB.1=0:TRISB.2=0: TRISB.3=0:TRISC.0=0:TRISC.1=0:TRISC.2=0:TRISC.4=1
'*****
'PUERTOS PARA COMUNICACION CON MODEM GSM
TX VAR PORTB.6:RX VAR PORTB.7
'*****
'VARIABLES PARA COMANDOS O NUMEROS DE CELULAR A PROGRAMAR
D0 VAR BYTE:D1 VAR BYTE:D2 VAR BYTE:D3 VAR BYTE:D4 VAR BYTE:D5 VAR BYTE
D6 VAR BYTE:D7 VAR BYTE:D8 VAR BYTE:D9 VAR BYTE:D10 VAR BYTE:TEMP1 VAR WORD
TEMP2 VAR WORD:TEMP3 VAR WORD:TEMP4 VAR WORD:clave VAR WORD:clave1 VAR WORD
modem VAR BYTE [8]:DATOS VAR BYTE [11]:numero1 VAR BYTE[9]
numero2 VAR BYTE[9]:numero3 VAR BYTE[9]:numero4 VAR BYTE[9]
'*****
'VARIABLES AUXILIARES DE PROGRAMA
A VAR BYTE:SIR VAR BIT:ARMAR VAR BIT:CONTA VAR BIT:TEMPO VAR BIT:TP VAR BIT
FV VAR BIT:POWER VAR BIT:CIRCULINA VAR BIT:CNT VAR BIT:A=4:CT VAR BIT
```

Figura 3.18. Configuración de LCD y Puertos analógicos como digitales y otras variables

En la Figura 3.19 se configuran las diferentes variables del programa como las variables para el sensor de temperatura DS18B20 y el GPS para leer la hora GMT.

```

'VARIABLES PARA SENSORES DS1820
temperature1 VAR WORD:temperaturex VAR WORD:temperatureY VAR WORD
temp VAR WORD:DQ1 VAR PORTB.4:SIGNO VAR BIT:SIG1 VAR BYTE:auto VAR BIT
'*****
HIGH SIRENA:LOW FOCO_ROJO:HIGH FOCO_VERDE:LOW CONTACTO1:LOW CONTACTO2
LOW CONTACTO3:LOW CONTACTO4
'*****
tip VAR WORD:SIG VAR BYTE:REF VAR BYTE
'*****
'Variables para lectura del GPS
GPR VAR BYTE [12] 'Se lee 5 bytes despues de recibir GPCCA,
'VARIABLES PARA LA HORA
Hora VAR BYTE: Hor VAR BYTE:HO VAR BYTE
'VARIABLES PARA MINUTOS
M VAR BYTE
S VAR BYTE 'VARIABLES PARA SEGUNDOS
HP1 VAR BYTE 'HORA PROGRAMADA
MP1 VAR BYTE 'MINUTO PROGRAMADO
HP2 VAR BYTE 'HORA PROGRAMADA
MP2 VAR BYTE 'MINUTO PROGRAMADO
'*****
'clave
WRITE 36,1:WRITE 37,2:WRITE 38,3:WRITE 39,4
READ 50,A:READ 51,SIG:READ 52,REF:READ 53,CT:READ 54,FV:READ 80,HP1:READ 81,MP1
READ 82,HP2:READ 83,MP2:READ 90,CNT:READ 91,auto
'*****

```

Figura 3.19 Variables del sensor de temperatura y Hora GMT en el GPS

En la Figura 3.20 se muestra el código necesario para la configuración del Modem GSM SIM900

```

'PROGRAMAR MODEM
SEROUT2 TX,84,["AT+IPR=9600",13]
PAUSE 2000
SEROUT2 TX,84,["AT+IFC=0,0",13]
SERIN2 RX,84,[WAIT ("OK")]
LCDOUT $fe, 1
LCDOUT "OK1"
SEROUT2 TX,84,["AT+CNMI=1,2,0,0,0",13]
SERIN2 RX,84,[WAIT ("OK")]
LCDOUT $fe, 1
LCDOUT "OK2"
SEROUT2 TX,84,["AT+CMGF=1",13]
SERIN2 RX,84,[WAIT ("OK")]
LCDOUT $fe, 1
LCDOUT "OK3"
SEROUT2 TX,84,["ATE0",13]
SERIN2 RX,84,[WAIT ("OK")]
LCDOUT $fe, 1
LCDOUT "OK4"
SEROUT2 TX,84,["AT&W",13]
SERIN2 RX,84,[WAIT ("OK")]
LCDOUT $fe, 1
LCDOUT "OK5"
PAUSE 2000

```

Figura 3.20 Configuración del MODEM GSM SIM 900

En la Figura 3.21 se registra el Modem GSM para Proveedor de Movistar o Claro según el chip que tenga instalado el modem.

```

REGISTRAR:
LCDOUT $fe, 1
LCDOUT "REGISTRANDO..."
SEROUT2 TX,84,["AT+CGREG?",13] ' PREGUNTA SI MODEN ESTA REGISTRADO CON ALGUNA OPERADORA
SERIN2 RX,84,1000,REGISTRAR,[WAIT ("+CGREG:"),STR modem \4]
PAUSE 100
IF MODEM[3]="1" THEN
    LCDOUT $fe, 1
    LCDOUT $fe, 2
    LCDOUT "MODEM REGISTRADO"
    PAUSE 1000
ELSE
    GOTO REGISTRAR
ENDIF
*****

```

Figura 3.21 Registro de Proveedor

En la Figura 3.22 se muestra el código donde el microcontrolador está a la espera de recibir un comando y si no recibe nada ejecuta la rutina que está en la etiqueta alarma cada 7 segundos, la cual se encarga de verificar el estado actual del contenedor en temperatura y energía eléctrica.

```

PRINCIPAL:
    GOSUB LEER_NUM
    GOSUB VER_NUM
Inicio:
    SERIN2 RX,84,7000,ALARMA,[WAIT("R"),STR DATOS \11] 'ESPERA POR COMANDOS
    'SI NO RECIBE COMANDOS SE EJECUTA RUTINAS QUE ESE ENCUENTRAN EN LA ETIQUETA ALARMA
INICIO2:
    D0=DATOS[0]
    D1=DATOS[1]
    D2=DATOS[2]
    D3=DATOS[3]
    D4=DATOS[4]
    D5=DATOS[5]
    D6=DATOS[6]
    D7=DATOS[7]
    D8=DATOS[8]
    D9=DATOS[9]
    D10=DATOS[10]
    *****
    'SE VISUALIZA EN LCD COMANDO RECIBIDO
    LCDOUT $fe, 1
    LCDOUT "Msg. Recibido!!!"
    LCDOUT $fe,$C0,D0,D1,D2,D3,D4,D5,D6,D7,D8,D9,D10
    PAUSE 2000

```

Figura 3.22 Programa Principal

En la Figura 3.23 se muestra la rutina cuando el modem recibe una llamada (RING). Esta rutina envía el estado del contenedor de temperatura y energía eléctrica.

```
'SELECCION DE CASOS
SELECT CASE D0
*****
CASE "I"
    TEMP1=D1-48
    TEMP1=TEMP1*1000
    TEMP2=D2-48
    TEMP2=TEMP2*100
    TEMP3=D3-48
    TEMP3=TEMP3*10
    TEMP4=D4-48
    clave1=TEMP1+TEMP2+TEMP3+TEMP4 'SE OBTIENE CLAVE
    clave1=1234
    IF clave=clave1 THEN ' SI LA CLAVE ES LA CORRECTA
        GOSUB LEER_SW ' RUTINA LEER ESTADO DE SW DE ARMAR
        GOSUB LEER_RTD ' RUTINA LEER EL SENSOR RTD
        GOSUB ENVIA_DATOS 'RUTINA QUE ENVIA DATOS POR SMS
        LCDOUT $fe, 1 ' RUTINA PARA INDICAR QUE SE ENVIO SMS
        LCDOUT "Msg. Enviado!!!"
        PAUSE 1000
    ENDIF
ENDIF
```

Figura 3.23 Comando RING (Recepción de llamada)

En la Figura 3.24 se muestra la rutina cuando el comando por mensaje de texto enviado es RTXnnnnnnnnnn donde x varia de 1 a 3 y nnnnnnnnnn es un numero de celular. Esta rutina programa los números de celulares a cual recibirá las alertas.

```
'PROGRAMAS NUMERO DE CELULAR QUE RECIBIRAN ESTADOS DE ALARMA
CASE "T" 'RTd1d2d3d4d5d6d7d8d9d10 ejemp: RT#968887721 comando =(RT1951551591)
    numero4[0]=D2-48:numero4[1]=D3-48:numero4[2]=D4-48
    numero4[3]=D5-48:numero4[4]=D6-48:numero4[5]=D7-48
    numero4[6]=D8-48:numero4[7]=D9-48:numero4[8]=D10-48
SELECT CASE D1
CASE "1" ' se programa telefono 1
    WRITE 0,numero4[0]:WRITE 1,numero4[1]:WRITE 2,numero4[2]
    WRITE 3,numero4[3]:WRITE 4,numero4[4]:WRITE 5,numero4[5]
    WRITE 6,numero4[6]:WRITE 7,numero4[7]:WRITE 8,numero4[8]
CASE "2" ' se programa telefono 2
    WRITE 9,numero4[0]:WRITE 10,numero4[1]:WRITE 11,numero4[2]
    WRITE 12,numero4[3]:WRITE 13,numero4[4]:WRITE 14,numero4[5]
    WRITE 15,numero4[6]:WRITE 16,numero4[7]:WRITE 17,numero4[8]
CASE "3" ' se programa telefono 3
    WRITE 18,numero4[0]:WRITE 19,numero4[1]:WRITE 20,numero4[2]
    WRITE 21,numero4[3]:WRITE 22,numero4[4]:WRITE 23,numero4[5]
    WRITE 24,numero4[6]:WRITE 25,numero4[7]:WRITE 26,numero4[8]

END SELECT
GOSUB LEER_NUM
```

Figura 3.24 Programación de números de celulares que recibirán alertas.

En la Figura 3.25 se programa el set point o alarma por temperatura

```
'PROGRAMAR SET POINT DE TEMPERATURA
CASE "S"
  IF D3="A" THEN
    TEMP2=D2-48
    TEMP4=TEMP2 'VALOR DEL SET POINT
  ELSE
    TEMP1=D2-48
    TEMP1=TEMP1*10
    TEMP2=D3-48
    TEMP4=TEMP1+TEMP2 'VALOR DEL SET POINT
  ENDIF
WRITE 51,D1 ' SE GUARDA EL SIGNO DEL SET POINT
WRITE 52,TEMP4 ' SE GUARDA EL VALOR DEL SET POINT
SIG=D1
REF=TEMP4
```

Figura 3.25 Rutina para programar alarma por temperatura

En la Figura 3.26 se muestra la rutina para armar la alarma, la cual tiene tres opciones Enviar un solo mensaje para el caso A, 3 mensajes para el caso B, para el caso E siempre envía mensajes cada cierto tiempo y ningún Mensaje en el caso C.

```
CASE "A" 'ARMAR ALARMA PARA ENVIO DE 1 SMS DE ALERTA , comando es: 'RccccXXXXXX
TEMP1=D1-48:TEMP1=TEMP1*1000:TEMP2=D2-48:TEMP2=TEMP2*100:TEMP3=D3-48
TEMP3=TEMP3*10:TEMP4=D4-48:clave1=TEMP1+TEMP2+TEMP3+TEMP4
IF clave=clave1 THEN
  A=0:FOCO_VERDE=0:HIGH SIRENA:WRITE 50,0:WRITE 54,1
ENDIF
CASE "B" 'ARMAR ALARMA PARA ENVIO DE 3 SMS DE ALERTA , comando es: 'RBccccXXXXXX
TEMP1=D1-48:TEMP1=TEMP1*1000:TEMP2=D2-48:TEMP2=TEMP2*100:TEMP3=D3-48
TEMP3=TEMP3*10:TEMP4=D4-48:clave1=TEMP1+TEMP2+TEMP3+TEMP4
IF clave=clave1 THEN
  A=3:tip=0:FOCO_VERDE=0:HIGH SIRENA:WRITE 50,3:WRITE 54,1
ENDIF
CASE "E" 'ARMAR ALARMA PARA ENVIO DE SMS CADA CIERTO TIEMPO DE ALERTA , comando es: 'REccccXXXXXX
TEMP1=D1-48:TEMP1=TEMP1*1000:TEMP2=D2-48:TEMP2=TEMP2*100:TEMP3=D3-48
TEMP3=TEMP3*10:TEMP4=D4-48:clave1=TEMP1+TEMP2+TEMP3+TEMP4
IF clave=clave1 THEN
  A=5:tip=0:FOCO_VERDE=0:HIGH SIRENA:WRITE 50,5:WRITE 54,1
ENDIF
CASE "C" 'ARMAR ALARMA PARA NO ENVIO DE SMS DE ALERTA , comando es: 'RCccccXXXXXX
TEMP1=D1-48:TEMP1=TEMP1*1000:TEMP2=D2-48:TEMP2=TEMP2*100:TEMP3=D3-48
TEMP3=TEMP3*10:TEMP4=D4-48:clave1=TEMP1+TEMP2+TEMP3+TEMP4
IF clave=clave1 THEN
  A=4:tip=0:FOCO_VERDE=1:HIGH SIRENA:WRITE 50,4:WRITE 54,0
ENDIF
```

Figura 3.26 Rutina para armado de alarma

En la Figura 3.27 y 3.28, se visualiza la rutina necesaria para activar los relés que activan o desactiva al contenedor.

```

CASE "W"
  TEMP1=D3-48:TEMP1=TEMP1*1000:TEMP2=D4-48:TEMP2=TEMP2*100:TEMP3=D5-48
  TEMP3=TEMP3*10:TEMP4=D6-48:clave1=TEMP1+TEMP2+TEMP3+TEMP4
  IF clave=clave1 THEN
    GOSUB escribir_dig 'RWHNccccXXXX,RWLNccccXXXX
  ENDIF

```

Figura 3.27 Rutina para determinar si la clave es correcta

```

escribir_dig:
  SELECT CASE D1
    CASE "H" 'RWH1ccccXXX COMANDO
      LOW CONTACTO1:LOW CONTACTO2:LOW CONTACTO3:LOW CONTACTO4
      CT=1
      WRITE 53,CT
    CASE "L" 'RWL1XXXXXXXXX COMANDO
      HIGH CONTACTO1:HIGH CONTACTO2:HIGH CONTACTO3:HIGH CONTACTO4
      CT=0
      WRITE 53,CT

  END SELECT

  RETURN

```

Figura 3.28 Rutina para ON / OFF los relés que controlan el inicio o fin del proceso en cada contenedor.

En la Figura 3.29 Se muestra la Rutina para obtener la hora y minuto de inicio de apagado del contenedor así como la hora de encendido. Estos datos son guardados en memoria

```

CASE "Z"
  TEMP1=D1-48
  TEMP1=TEMP1*10
  TEMP2=D2-48
  TEMP3=TEMP1+TEMP2
  HP1=TEMP3
  WRITE 80,HP1
  TEMP1=D3-48
  TEMP1=TEMP1*10
  TEMP2=D4-48
  TEMP3=TEMP1+TEMP2
  MP1=TEMP3
  WRITE 81,MP1
  TEMP1=D5-48
  TEMP1=TEMP1*10
  TEMP2=D6-48
  TEMP3=TEMP1+TEMP2
  HP2=TEMP3
  WRITE 82,HP2
  TEMP1=D7-48
  TEMP1=TEMP1*10
  TEMP2=D8-48
  TEMP3=TEMP1+TEMP2
  MP2=TEMP3
  WRITE 83,MP2
  CNT=0
  WRITE 90,CNT
  LCDOUT $fe, 1, DEC2 Hora,":",DEC2 M,":", DEC2 S
  LCDOUT $FE,$C0, DEC2 HP1,":",DEC2 MP1," ",DEC2 HP2,":",DEC2 MP2
  PAUSE 2000

```

Figura 3.29 Rutina de Hora y minuto para encendido y apagado de contenedor

En la Figura 3.30 se muestra la rutina para obtener la hora GMT de los datos que envía el GPS, y luego determinar si se enciende o apaga el contenedor.

```

Tiempo:
SERIN2 PORIC.4,84,3000,PRINCIPAL,{WAIT("GPRMC,"),STR GPR\12] 'GPGGA GPRMC
' SE OBTIENE HORA MINUTOS Y SEGUNDOS
HO=(GPR[0]-48)*10+GPR[1]-48
IF HO <= 4 THEN
    Hor = HO + 24
    Hora = Hor - 5
ELSE
    Hora = HO - 5
ENDIF
M = (GPR[2]-48)*10+GPR[3]-48:S = (GPR[4]-48)*10+GPR[5]-48
IF auto=1 THEN
    IF HORA = HP1 AND CNT=0 THEN
        IF M=MP1 THEN
            HIGH CONTACTO1:HIGH CONTACTO2:HIGH CONTACTO3:HIGH CONTACTO4
            CT=0:CNT=1:WRITE 53,CT:WRITE 90,CNT
        ENDIF
    ENDIF
    IF HORA = HP2 AND CNT=1 THEN
        IF M>=MP2 THEN
            LOW CONTACTO1:LOW CONTACTO2:LOW CONTACTO3:LOW CONTACTO4
            CT=1:CNT=0:WRITE 53,CT:WRITE 90,CNT
        ENDIF
    ENDIF
ENDIF
RETURN

```

Figura 3.30 Rutina para leer hora del GPS y encender o apagar contenedor

En la Figura 3.31 se muestra la rutina donde se determina si el encendido y apagado es de manera automática teniendo en cuenta la hora del GPS o manual que sería decisión del usuario cuando apaga o enciende el contenedor.

```

CASE "Y"
    TEMP1=D1-48
    TEMP1=TEMP1*1000
    TEMP2=D2-48
    TEMP2=TEMP2*100
    TEMP3=D3-48
    TEMP3=TEMP3*10
    TEMP4=D4-48
    clave1=TEMP1+TEMP2+TEMP3+TEMP4
    TEMP1=D5-48
    IF clave=clave1 THEN

        IF TEMP1=1 THEN
            auto=1
            WRITE 91,auto
        ELSE
            auto=0
            WRITE 91,0
        ENDIF
    ENDIF

```

Figura 3.31 Rutina que determina si el encendido y apagado del contenedor es automático o manual.

En la Figura 3.32 se muestra la rutina para el envío de los mensajes los datos de temperatura y energía eléctrica del contenedor este mensaje se envía cuando el modem recibe una llamada (RING) .

```

ENVIA_DATOS:
    'ENVIA DATOS SOLICITADOS CON COMANDO RING
    'ENVIA ESTADO SWITCH DE ARMADO, CONTENEDOR , SENSOR DE TEMPERATURA
    SEROUT2 TX,84,["Armar=";34,DEC numero1{0},DEC numero1{1},DEC numero1{2},DEC numero1{3},DEC numero1{4}
    PAUSE 1000
    SEROUT2 TX,84,["ARMAR: ",DEC1 ARMAR," SIRE="; BIN SIRE," DOUT: ",DEC1 POWER," TEMPERATURA:",STR SIRE1 \1,
    PAUSE 5000
    RETURN
  
```

Figura 3.32 Rutina para envío de mensajes de alertas

En la Figura 3.33 se muestra la rutina ALARMA, esta rutina se ejecuta cada 7 segundos. La cual se encarga de leer el estado eléctrico del contenedor si esta encendido o apagado así como verifica si se habilito (ARMO) alarma (Figura 3.34). También ejecuta la rutina Tiempo(Figura 3.35) que se encarga de leer la hora del GPS y procede al encendido o apagado del contenedor, ejecuta la rutina de leer la temperatura que envía el sensor DS18B20. Por ultimo si se la temperatura q se lee no está dentro de los parámetros permitido se genera una alarma y se envía un mensaje de alerta de igual manera si el contenedor se apaga de manera inesperada.

```

ALARMA:
    'HIGH SIRENA
    GOSUB LEER_SW ' LEER SW QUE HABILITA O ARMA LA ALARMA
    GOSUB Tiempo
    GOSUB LEER_RTD 'LEER SENSOR DE TEMPERATURA

    IF ARMAR=1 THEN
        IF POWER=0 OR IP=1 THEN
            CIRCULINA=1
            LOW SIRENA
            GOSUB MENSAJE_Aviso ' ENVIA MENSAJE DE ALERTA A 3 NUMEROS CONFIGURADOS

        ELSE
            CIRCULINA=0
            HIGH SIRENA
        ENDIF
    ELSE
        CIRCULINA=0
    ENDIF

    GOTO INICIO
  
```

Figura 3.33 Rutina ALARMA

```

LEER_SW:
    IF PORTD.1=0 AND FV=1 THEN
        FOCO_VERDE=1
        FV=0
        A=4
        tip=0
        WRITE 50,4
        WRITE 54,FV
        HIGH SIRENA
        PAUSE 2000
    ENDIF
    IF PORTD.1=0 AND FV=0 THEN
        FOCO_VERDE=0
        FV=1
        A=5
        WRITE 50,5
        WRITE 54,FV
        PAUSE 2000
    ENDIF
    ARMAR=~FOCO_VERDE
    SIR=~PORTB.3
    POWER=PORTD.0
RETURN
*****

```

Figura 3.34 Rutina leer habilitación de alarma y verificar ON/OFF contenedor

```

'LEER SENSOR RTD
LEER_RTD:
OWOUT DQ1, 1, [SCC, $44]:PAUSE 200
OWOUT DQ1, 1, [SCC, $BE]:OWIN DQ1, 0, [temperature1.LOWBYTE, temperature1.HIGHBYTE]
SIGNO = temperature1.14

```

Figura 3.35. Parte del código de lectura de temperatura del sensor DS18B20

3.4 DESCRIPCIÓN DEL PROGRAMA PARA CELULAR

Para el proyecto se ha desarrollado una aplicación para plataforma ANDROID, ha sido desarrollado utilizando el APPINVENTOR 2. La aplicación debe cumplir con las siguientes características principales:

- Programar los tres números de celular que recibirán las alertas
- Solicitar por llamada telefónica solo timbrando al número del modem GSM y recibir la temperatura del contenedor y si el contenedor tiene o no energía eléctrica.
- Ingresar el valor de referencia de la temperatura para las alarmas
- Permitir selección de modo manual o automático para el encendido y apagado del contenedor
- Armar y Desarmar alarmas o alertas por SMS
- Programar la hora de encendido y apagado del contenedor

Las pantallas de esta aplicación se muestra en la Figura 3.36



Figura 3.36. Pantallas de la aplicación

Para el desarrollo de esta aplicación se usa el APPINVENTOR 2, En la Figura 3.37 y Figura 3.38 se muestran las ventanas en modo diseño de la aplicación, donde se aprecia cómo están distribuidos los elementos y/o componentes de la aplicación.

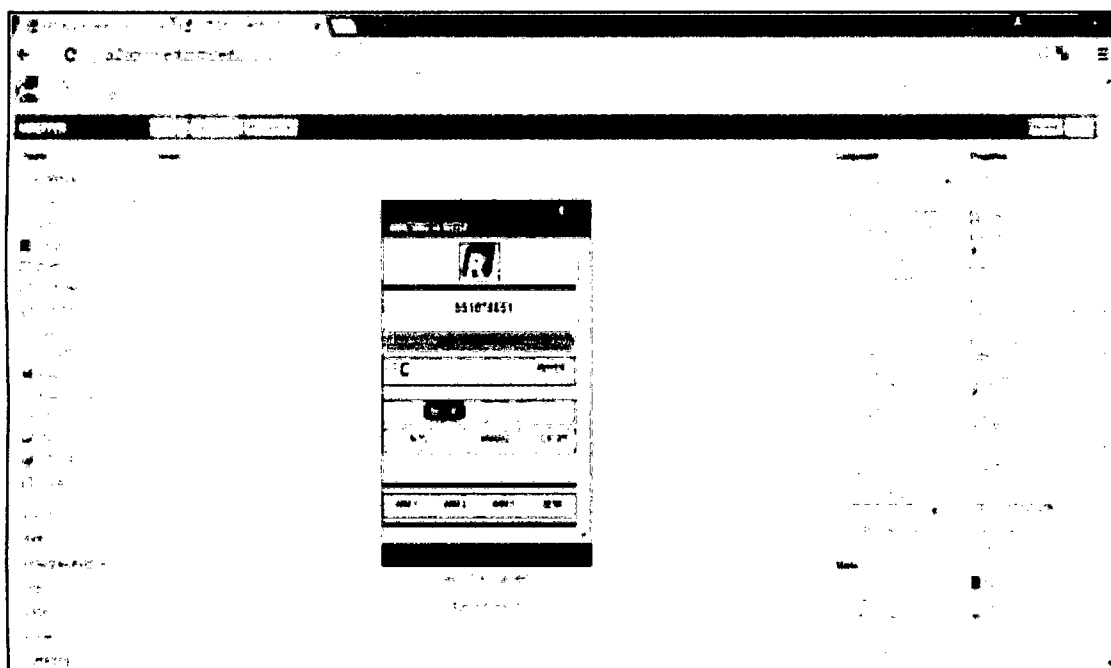


Figura 3.37 Ventana modo diseño de la aplicación

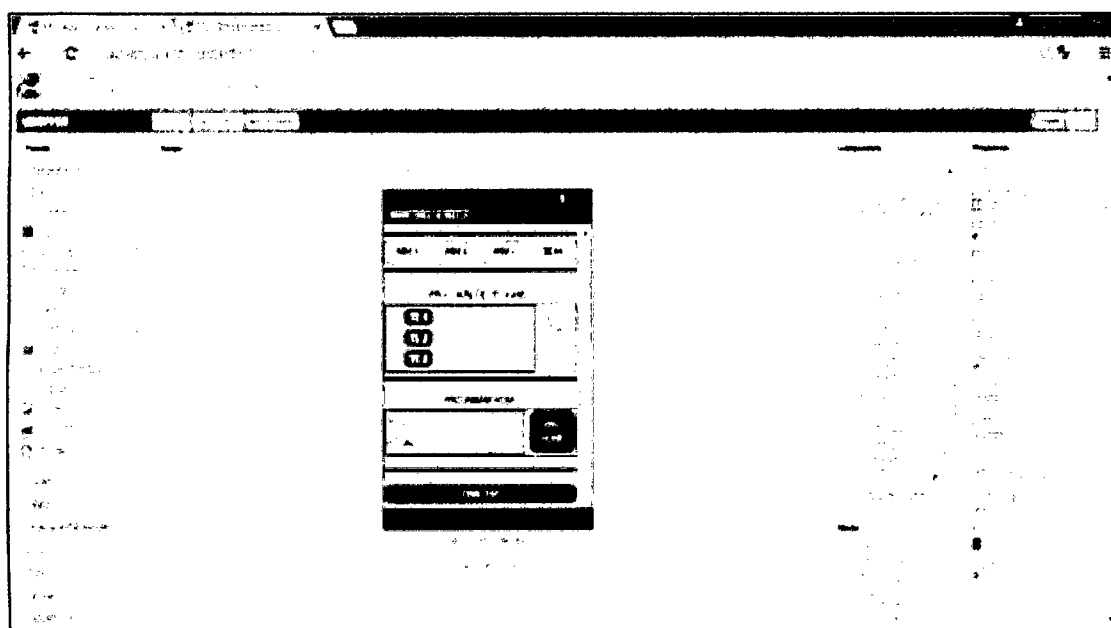


Figura 3.38 Ventana modo diseño de la aplicación

Los componentes que se usan para esta aplicación son los que se muestran en la Figura 3.39, 3.40, 3.41.

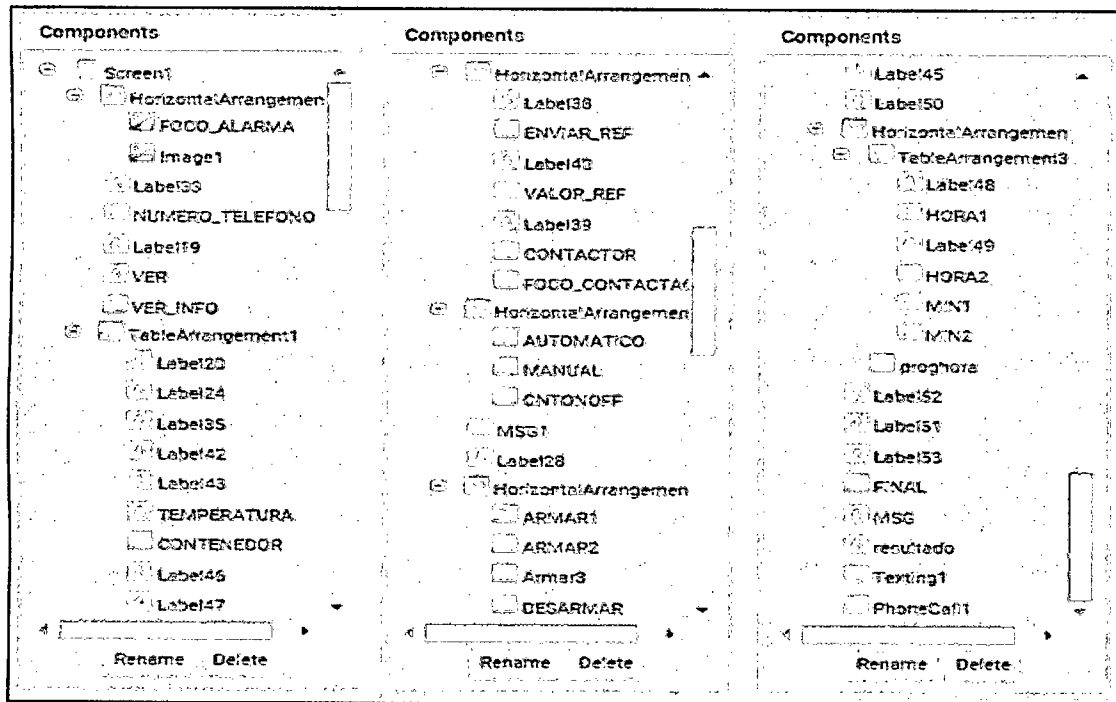


Figura 3.39 Lista de componentes de la aplicación

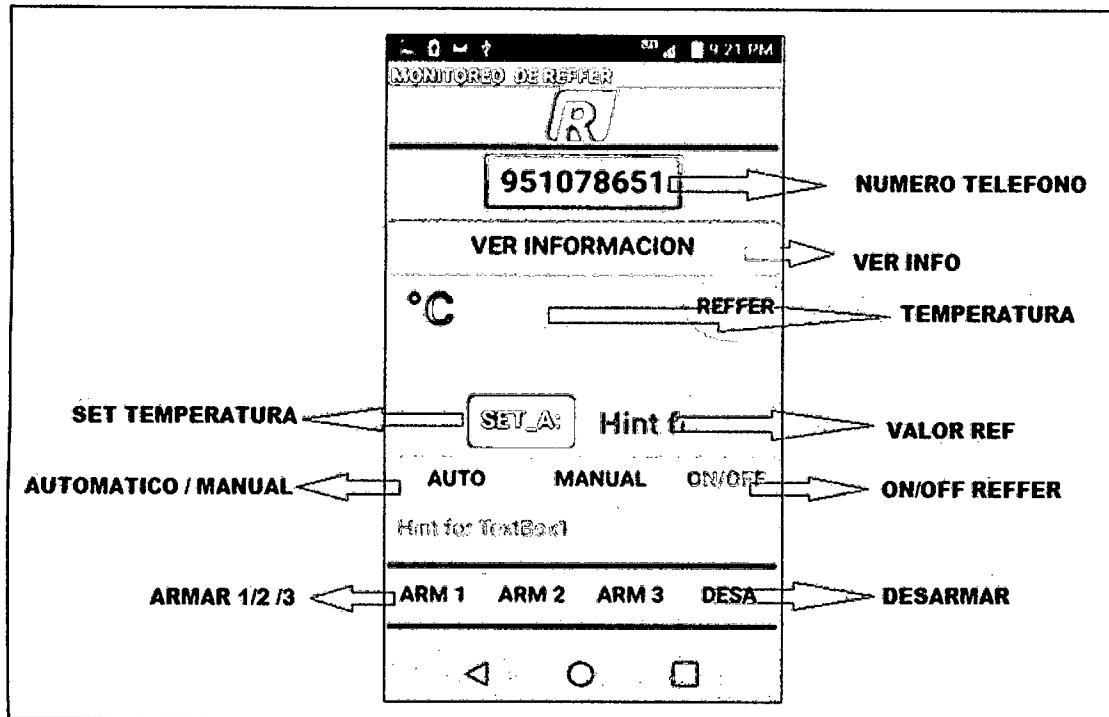


Figura 3.40 Nombre de componentes de la aplicación administrador

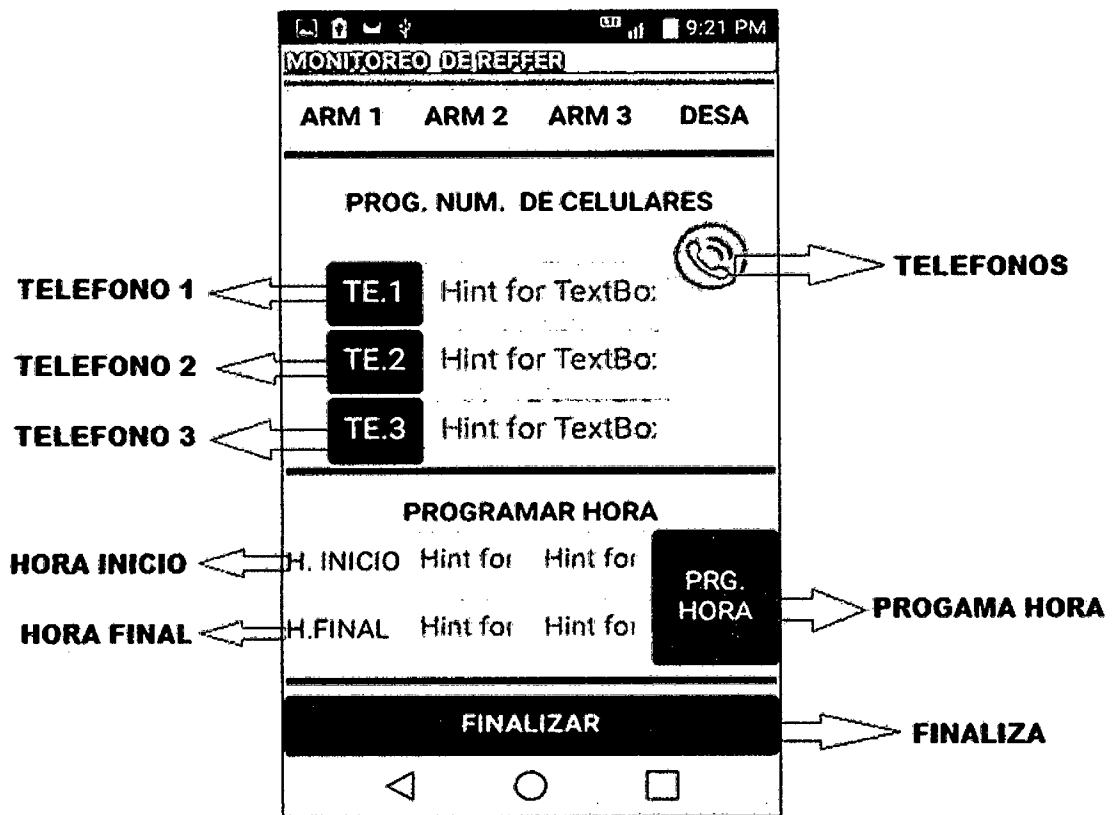


Figura 3.41 Nombre de componentes de la aplicación administrador

3.4.1 DESCRIPCIÓN DEL CÓDIGO DE PROGRAMA DE APLICACIÓN PARA ADMINISTRADOR

En la Figura 3.42 se muestra los bloques de los diferentes programas por componentes y/o eventos, los cuales se describirán a continuación.

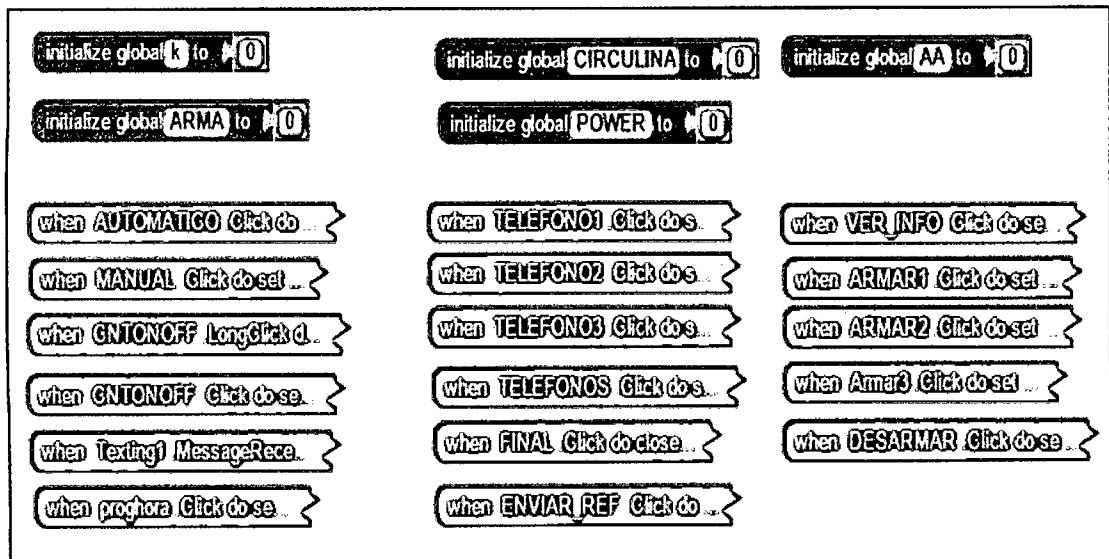


Figura 3.42 Bloques de la aplicación para administrador

3.4.2 BOTÓN VER_INFO

Al presionar (click) este botón se realiza una llamada al número telefónico ingresado en la caja de texto NUMERO_TELEFONO, este número telefónico corresponde al chip instalado en el modem GSM SIM900 y a continuación se recibe un mensaje de texto con el estado de energía eléctrica del contenedor y la temperatura. Este mensaje se muestra en la etiqueta llamada MSG1 como se puede visualizar en la Figura 3.44

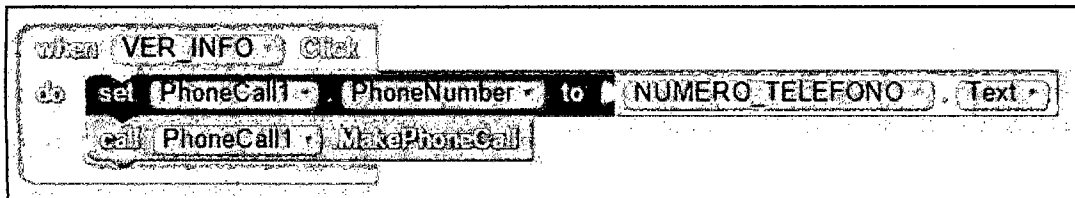


Figura 3.43 Código para botón VER_INFO

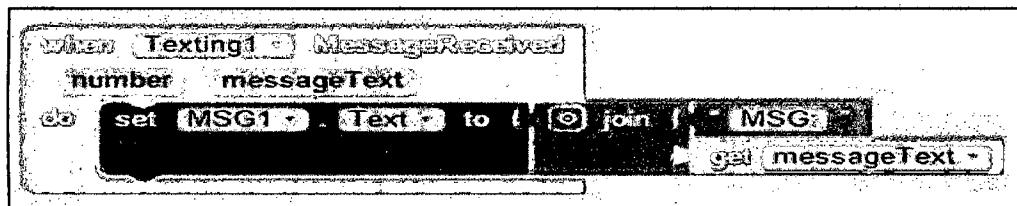


Figura 3.44 Código para Recepción de mensajes de texto

3.4.3 BOTÓN TELEFONO1

Al presionar (click) este botón se envía un mensaje de texto SMS con el formato "RT1NNNNNNNNNN" donde "NNNNNNNNNN" es un numero de celular y corresponde el celular número 1, que se ingresa por la caja de texto N_TELEFONO1, este SMS se envía al número telefónico ingresado en la caja de texto NUMERO_TELEFONO, este número telefónico corresponde al chip instalado en el modem GSM SIM900. Ha este número telefónico se podrá enviar las alertas y el valor de la temperatura y si el contenedor esta encendido o apagado.

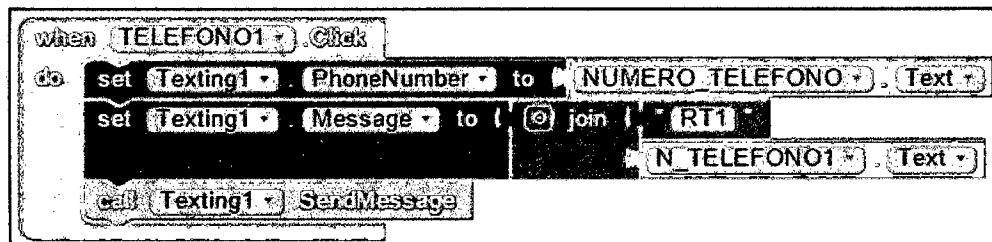


Figura 3.45 Código para TELEFONO1

3.4.4 BOTÓN TELEFONO2

Al presionar (click) este botón se envía un mensaje de texto SMS con el formato "RT2NNNNNNNNNN" donde "NNNNNNNNNN" es un numero de celular y

corresponde el celular número 2 que se ingresa por la caja de texto N_TELEFONO2, este SMS se envía al número telefónico ingresado en la caja de texto NUMERO_TELEFONO, este número telefónico corresponde al chip instalado en el modem GSM SIM900. Ha este número telefónico se podrá enviar las alertas.

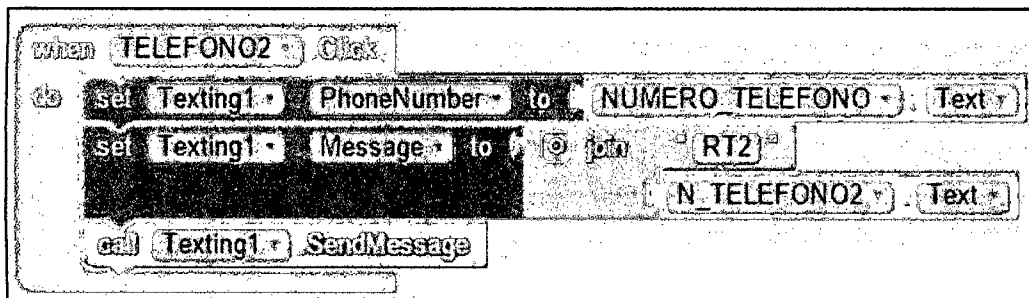


Figura 3.46 Código para TELEFONO2

3.4.5 BOTÓN TELEFONO3

Al presionar (click) este botón se envía un mensaje de texto SMS con el formato "RT3NNNNNNNNNN" donde "NNNNNNNNNN" es un numero de celular y corresponde el celular número 3 que se ingresa por la caja de texto N_TELEFONO3, este SMS se envía al número telefónico ingresado en la caja de texto NUMERO_TELEFONO, este número telefónico corresponde al chip instalado en el modem GSM SIM900. Ha este número telefónico se podrá enviar las alertas.

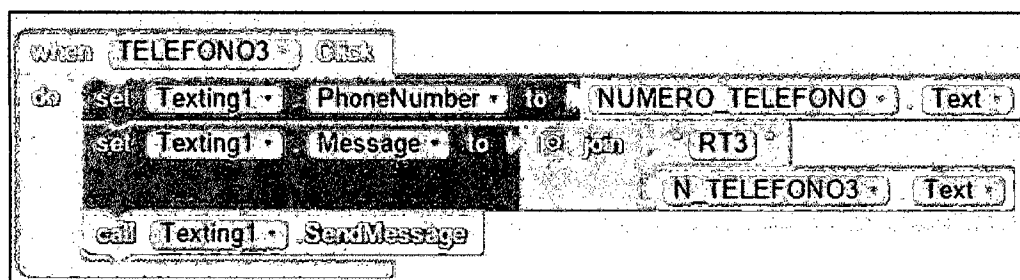


Figura 3.47 Código para TELEFONO3

3.4.6 BOTÓN TELEFONOS

Al presionar (click) este botón se envía un mensaje de texto SMS con el formato “RDCCCC123456” donde “CCCC” es la clave de 4 dígitos, este SMS se envía al número telefónico ingresado en la caja de texto NUMERO_TELEFONO, este número telefónico corresponde al chip instalado en el modem GSM SIM900 y a continuación se recibe un mensaje de texto con los números de celulares programados en el microcontrolador. Este mensaje se muestra en la etiqueta llamada MSG1.

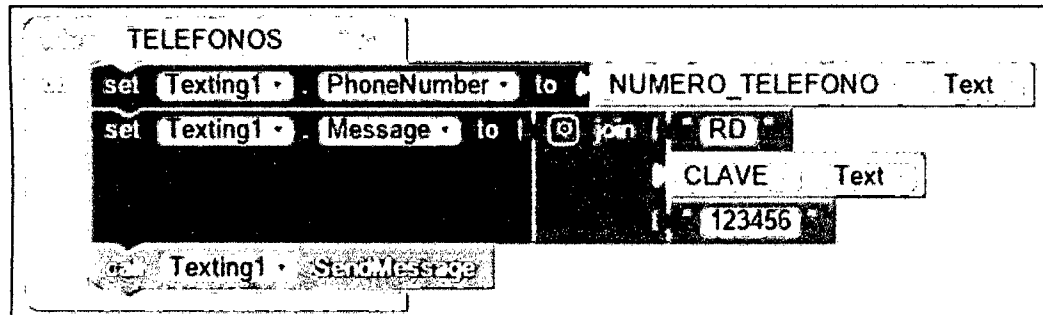


Figura 3.48 Código para TELEFONOS

3.4.7 BOTÓN ARMAR1

Al presionar (click) este botón se envía un mensaje de texto SMS con el formato “RACCCC123456” donde “CCCC” es la clave de 4 dígitos, este SMS se envía al número telefónico ingresado en la caja de texto NUMERO_TELEFONO, este número telefónico corresponde al chip instalado en el modem GSM SIM900. Este comando hace que el microcontrolador envíe un solo SMS de alerta a los celulares programados.

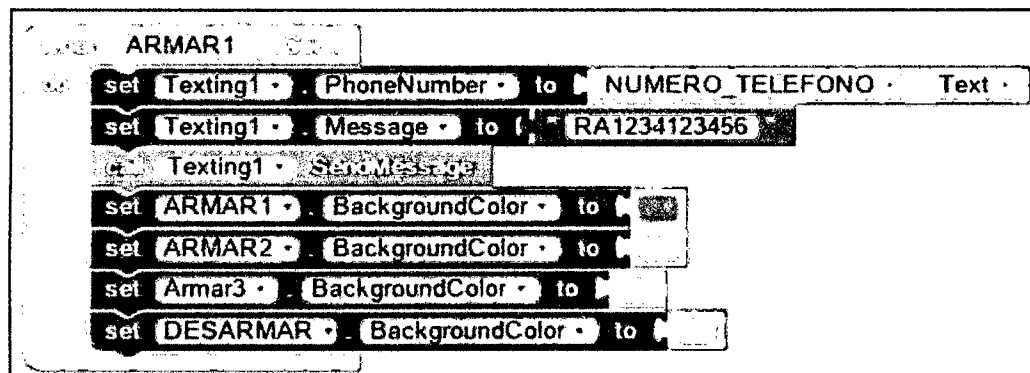


Figura 3.49 Código para ARMAR1

3.4.8 BOTÓN ARMAR2

Al presionar (click) este botón se envía un mensaje de texto SMS con el formato “RBCCCC123456” donde “CCCC” es la clave de 4 dígitos, este SMS se envía al número telefónico ingresado en la caja de texto NUMERO_TELEFONO, este número

telefónico corresponde al chip instalado en el modem GSM SIM900. Este comando hace que el microcontrolador envíe tres SMS de alerta a los celulares programados.

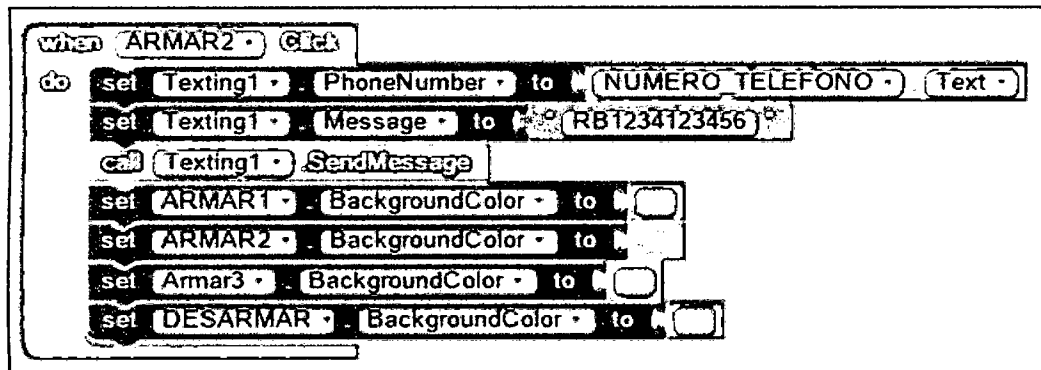


Figura 3.50 Código para ARMAR2

3.4.9 BOTÓN ARMAR3

Al presionar (click) este botón se envía un mensaje de texto SMS con el formato “RECCCC123456” donde “CCCC” es la clave de 4 dígitos, este SMS se envía al número telefónico ingresado en la caja de texto NUMERO_TELEFONO, este número telefónico corresponde al chip instalado en el modem GSM SIM900. Este comando hace que el microcontrolador envíe SMS de alerta de manera continua y cada cierto tiempo a los celulares programados.

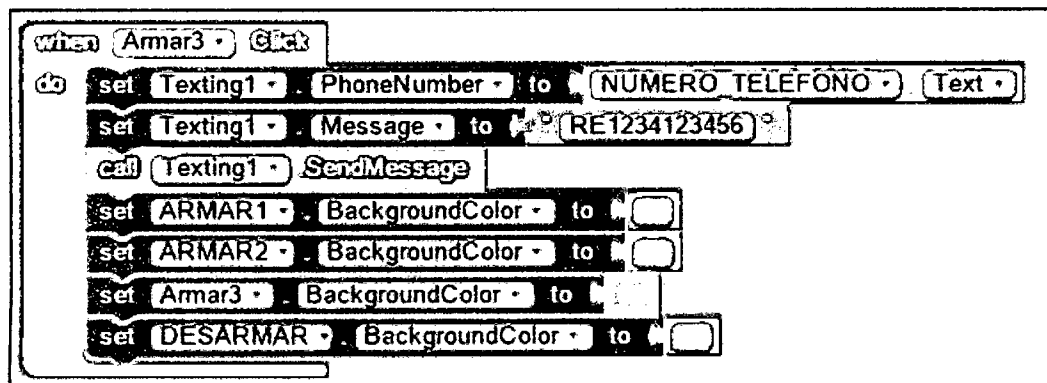


Figura 3.51 Código para ARMAR2

3.4.10 BOTÓN DESARMAR

Al presionar (click) este botón se envía un mensaje de texto SMS con el formato “RCXXXX123456” donde “XXXX” es la clave de 4 dígitos, este SMS se envía al número telefónico ingresado en la caja de texto NUMERO_TELEFONO, este número telefónico corresponde al chip instalado en el modem GSM SIM900. Este comando hace que el microcontrolador no envíe SMS de alerta a los celulares programados

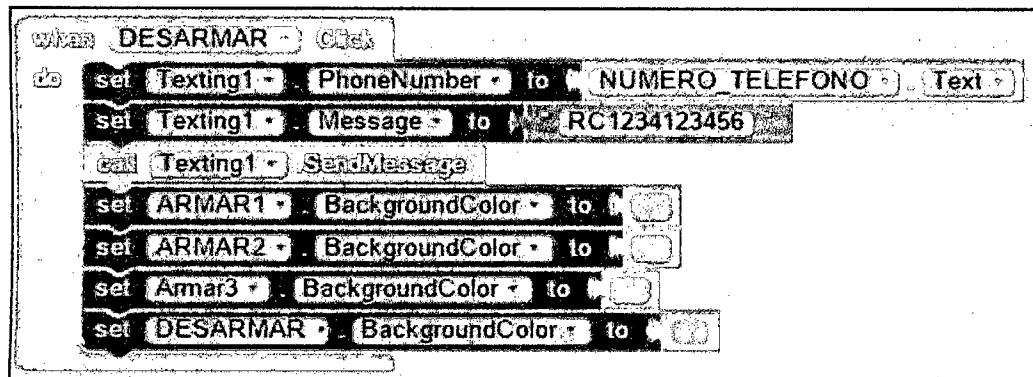


Figura 3.52 Código para DESARMAR

3.4.11 BOTÓN AUTOMATICO

Al presionar (click) este botón se envía un mensaje de texto SMS con el formato “RY1234123456”, este SMS se envía al número telefónico ingresado en la caja de texto NUMERO_TELEFONO, este número telefónico corresponde al chip instalado en el modem GSM SIM900. Este comando hace que el contenedor se pueda encender o apagar de manera AUTOMATICA programando la hora de apagado y encendido del contenedor.

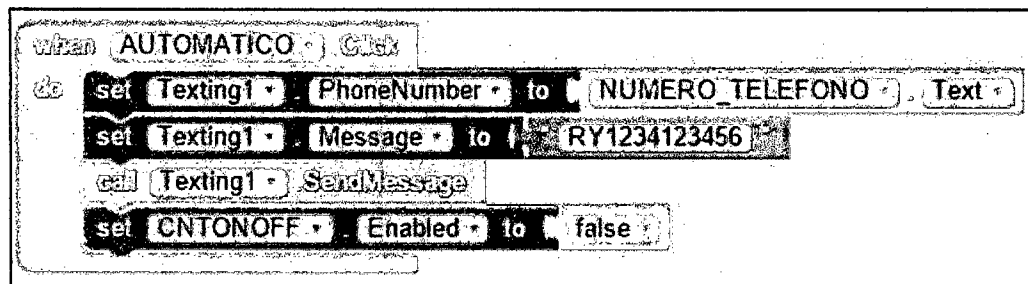


Figura 3.53 Código para AUTOMATICO

3.4.12 BOTÓN MANUAL

Al presionar (click) este botón se envía un mensaje de texto SMS con el formato “RY1234223456”, este SMS se envía al número telefónico ingresado en la caja de texto NUMERO_TELEFONO, este número telefónico corresponde al chip instalado en el modem GSM SIM900. Este comando hace que el contenedor se pueda encender o apagar de manera manual y por el usuario encargado de supervisar o monitorear el contenedor

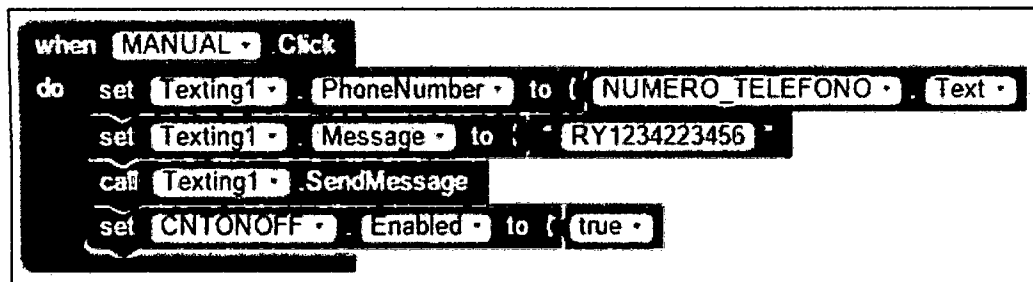


Figura 3.54 Código para MANUAL

3.4.13 BOTÓN CNTONOFF.CLICK

Al presionar (click) este botón se envía un mensaje de texto SMS con el formato “RWH112341234”, este SMS se envía al número telefónico ingresado en la caja de texto NUMERO_TELEFONO, este número telefónico corresponde al chip instalado en el modem GSM SIM900. Este comando hace que el microcontrolador active 4 relés que a la vez enciende el contenedor.

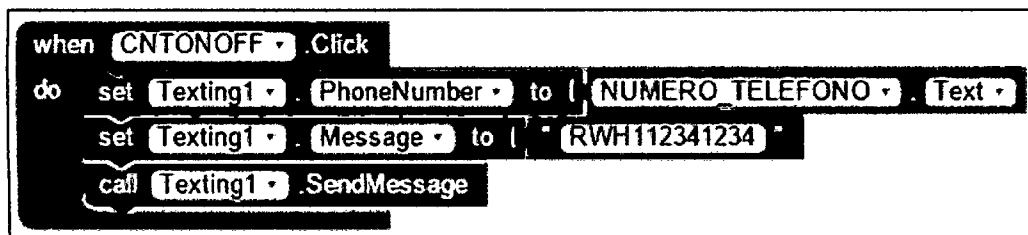


Figura 3.55 Código para CNTONOFF.CLICK

3.4.14 BOTÓN CNTONOFF.LONGCLICK

Al presionar (click) este botón se envía un mensaje de texto SMS con el formato “RWL112341234”, este SMS se envía al número telefónico ingresado en la caja de texto NUMERO_TELEFONO, este número telefónico corresponde al chip instalado en el modem GSM SIM900. Este comando hace que el microcontrolador desactive 4 relés que a la vez apaga el contenedor.

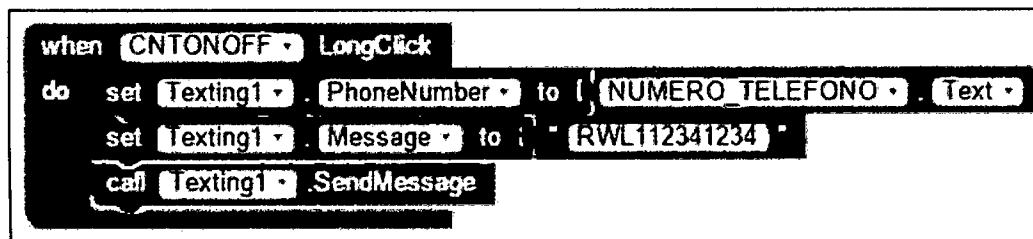


Figura 3.56 Código para CNTONOFF.LONGCLICK

3.4.15 BOTÓN ENVIAR_REF

Al presionar (click) este botón se envía un mensaje de texto SMS con el formato “RSVALORA5678912”, este SMS se envía al número telefónico ingresado en la caja de texto NUMERO_TELEFONO, este número telefónico corresponde al chip instalado en el modem GSM SIM900. Este comando envía el valor de referencia de la temperatura a la cual se van a generar las alarmas.

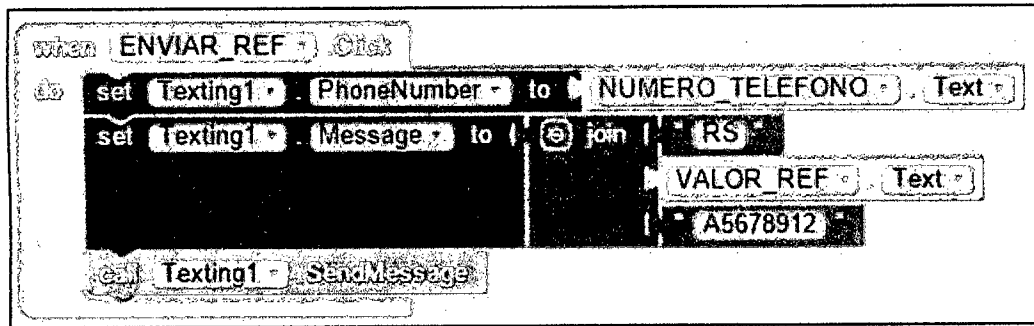


Figura 3.57 Código para enviar referencia de temperatura para alarma

3.4.16 BOTÓN proghora

Al presionar (click) este botón se envía un mensaje de texto SMS con el formato “RZHHMMMhhmm123456”, este SMS se envía al número telefónico ingresado en la caja de texto NUMERO_TELEFONO, este número telefónico corresponde al chip instalado en el modem GSM SIM900. Este comando envía la hora de inicio o que se apaga el contenedor hora y minuto de dos dígitos (HHMM) y la hora y minuto (hhmm) de dos dígitos que viene hacer la hora que se volverá encender.

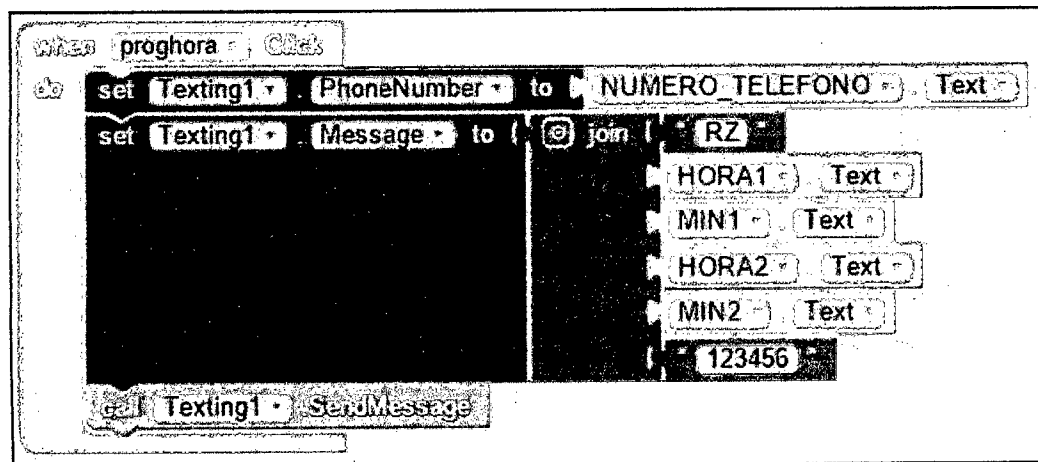


Figura 3.58 Código para enviar la hora de apagado y encendido del contenedor

3.4.17 BOTÓN FINALIZAR

Al hacer click la aplicación se cerrara y culminara programa

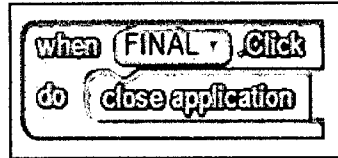


Figura 3.59 FINALIZAR APLICACION

En este capítulo se mostrara la tabla de costos para la implementación del proyecto en una se mostrara solo los costos de materiales y la otra tabla se considera los costos de ingeniería con un costo por hora de S/.40.

4.1 COSTO DE MATERIALES

Tabla 4.1 Costo de materiales

ITEM	MATERIALES	CANT.	P.UNIT.	P. TOTAL
1	MICROONTROLADOR PIC16F877A	1	S/. 20.00	S/. 20.00
2	LCD 16 X2	1	S/. 22.00	S/. 22.00
3	FUENTE 5V INDUSTRIAL	1	S/. 190.00	S/. 190.00
4	FUENTE 5V NORMAL	1	S/. 25.00	S/. 25.00
5	MODULO 4 RELES	1	S/. 35.00	S/. 35.00
6	MODULO 2 RELES	1	S/. 15.00	S/. 15.00
7	RELAY 11 PINES	1	S/. 20.00	S/. 20.00
8	SENSOR DS18B20	1	S/. 20.00	S/. 20.00
9	MODULO GSM SIM900	1	S/. 170.00	S/. 170.00
10	MODULO GPS SKYLAB	1	S/. 90.00	S/. 90.00
11	RESISTENCIAS	10	S/. 0.30	S/. 3.00
12	LEDS	5	S/. 0.40	S/. 2.00
13	CRISTAL	1	S/. 2.50	S/. 2.50
14	PULSADORES	2	S/. 3.00	S/. 6.00
15	CIRCULINA	1	S/. 50.00	S/. 50.00
16	PLACA IMPRESA	1	S/. 100.00	S/. 100.00
17	CAJA METALICA PARA PLACA	1	S/. 125.00	S/. 125.00
18	CONECTORES/CABLES	1	S/. 30.00	S/. 30.00
	TOTAL			S/. 925.50

4.2 COSTO DE MATERIALES MÁS INGENIERIA

Tabla 4.2. Costos de materiales más Ingeniería

ITEM	MATERIALES	CANT.	P.UNIT.	P. TOTAL
1	MICROONTROLADOR PIC16F877A	1	S/. 20.00	S/. 20.00
2	LCD 16 X2	1	S/. 22.00	S/. 22.00
3	FUENTE 5V INDUSTRIAL	1	S/. 190.00	S/. 190.00
4	FUENTE 5V NORMAL	1	S/. 25.00	S/. 25.00
5	MODULO 4 RELES	1	S/. 35.00	S/. 35.00
6	MODULO 2 RELES	1	S/. 15.00	S/. 15.00
7	RELAY 11 PINES	1	S/. 20.00	S/. 20.00
8	SENSOR DS18B20	1	S/. 20.00	S/. 20.00
9	MODULO GSM SIM900	1	S/. 170.00	S/. 170.00
10	MODULO GPS SKYLAB	1	S/. 90.00	S/. 90.00
11	RESISTENCIAS	10	S/. 0.30	S/. 3.00
12	LEDS	5	S/. 0.40	S/. 2.00
13	CRISTAL	1	S/. 2.50	S/. 2.50
14	PULSADORES	2	S/. 3.00	S/. 6.00
15	CIRCULINA	1	S/. 50.00	S/. 50.00
16	PLACA IMPRESA	1	S/. 100.00	S/. 100.00
17	CAJA METALICA PARA PLACA	1	S/. 125.00	S/. 125.00
18	CONECTORES/CABLES	1	S/. 30.00	S/. 30.00
19	INGENIERIA COSTO POR HORA	40	S/. 50.00	S/. 2,000.00
	TOTAL			S/. 2,925.50

5.0 CONCLUSIONES Y RECOMENDACIONES

5.1 CONCLUSIONES

Podemos concluir y afirmar el diseño y la construcción de un nuevo sistema de monitoreo con tecnología GSM/GPS y ahorro de energía para contenedores de producto congelados

5.2 RECOMENDACIONES

La utilización de nuevos chips y sistemas de comunicación para motivar el uso de nuevas tecnologías para el desarrollo de equipos electrónicos.

Incentivar, motivar e investigar las nuevas tecnologías para la solución de problemas aplicando electrónica e informática.